Personalized Recommendations of Locally Interesting Venues to Tourists via Cross Region Community Matching

YI-LIANG ZHAO, National University of Singapore
LIQIANG NIE, National University of Singapore
XIANGYU WANG, National University of Singapore
TAT-SENG CHUA, National University of Singapore

You are in a new city. You are not familiar with the places and neighborhoods. You want to know all about the exciting sights, food outlets and cultural venues that the locals frequent, in particular those that suit your personal interests. Even though there exist many mapping sites, local search and travel assistance sites, they mostly provide popular and famous listings such as Statue of Liberty and Eiffel Tower, which are well-known places but may not suit your personal needs or interests. Therefore, there is a gap between what tourists want and what dominant tourism resources are providing. In this work, we seek to provide a solution to bridge this gap by exploiting the rich user generated location contents in location-based social networks in order to offer tourists the most relevant and personalized local venue recommendations. In particular, we first propose a novel Bayesian approach to extract the social dimensions of people at different geographical regions to capture their latent local interests. We next mine the local interest communities in each geographical region. We then represent each local communities using aggregated behaviours of community members. Finally, we correlate communities across different regions and generate venue recommendations to tourists via cross region community matching. We have sampled a representative subset of check-ins from Foursquare and experimentally verified the effectiveness of our proposed approaches.

Categories and Subject Descriptors: H.3.3 [**Information Search and Retrieval**]: Information Filtering; J.4 [**Social and Behavioral Sciences**]: Sociology

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Location-based Social Networks, Social Dimensions, Locally Interesting Venues, Cross Region Community Matching

ACM Reference Format:
Yi-Liang Zhao, Liqiang Nie, Xiangyu Wang and Tat-Seng Chua, 2013. Personalized Recommendations of Locally Interesting Venues to Tourists via Cross Region Community Matching. *ACM Trans. Intell. Syst. Technol.* 9, 4, Article 1 (September 2013), 26 pages.
DOI:http://dx.doi.org/10.1145/0000000.0000000

1. INTRODUCTION

When we travel to new places, in addition to sightseeing, we are often interested in exploring local cultures, which match our personal interests, such as sampling local cuisines, understanding local customs, and visiting shops selling local special items, etc. However, there exists a large gap between what we want and what we are provid-

---

This work was supported by NUS-Tsinghua Extreme Search project under the grant number: R-252-300-001-490 and A*STAR Project Geographical Information Retrieval via Spatial Annotation of Web Media.
Author's addresses: Y.-L. Zhao (corresponding author), National University of Singapore, Singapore; email: zhaoyiliang@gmail.com; L. Nie, X. Wang, T.-S. Chua, National University of Singapore, Singapore.

DOI:http://dx.doi.org/10.1145/0000000.0000000

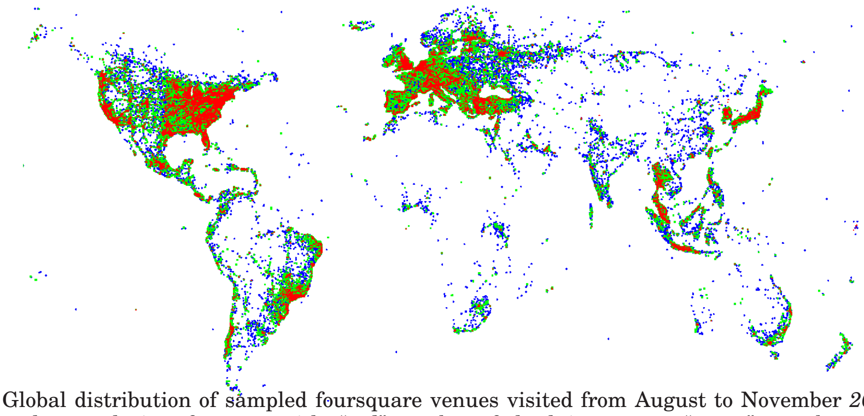


Fig. 1. Global distribution of sampled foursquare venues visited from August to November *2012*. Colors represent the popularity of venues with: "red"-number of check-ins > 100, "green"- number of check-ins > 50 and "blue"- number of check-ins > 10.

ed by the dominant tourism resources, such as Wikitravel[1], Lonely Planet[2] and official tourism boards of certain countries, such as YourSingapore[3] and AustraliaTravel[4]. The gap is caused by two main reasons. First, these sites mainly provide information of famous attractions or popular local landmarks instead of locally interesting places. However, many tourists may want to experience local cultures that match their interests in terms of local food, events and shops. These locally interesting places or activities may not be famous enough to be included in these tourism resources. Second, the dominant tourism resources generate user-independent contents while people usually have drastically different personal preferences in reality. For example, people who love shopping may want to visit more popular local shops, food lovers are more interested in sampling different kinds of local foods, such as the local foods in Shilin Night Market in Taipei and people who enjoy night life will be happier to be recommended with local bars and pubs.

On the other hand, rich location data at fine-grained level is now available from the recently emerging location-based social networks (LBSNs), such as Foursquare and Gowalla. LBSNs are becoming more and more popular thanks to the recent availability of open mobile platforms, which make LBSNs much more accessible to mobile users. These LBSNs are able to provide sufficient resources to bridge the aforementioned gap. First, they allow users to voluntarily annotate the real world with check-ins[5] which indicate the specific times that the users were at particular locations. Fig. 1 shows the sampled distribution of Foursquare venues visited during August to November, *2012*. The high density check-in distribution in such a short period reveals the worldwide active participation of Foursquare users. In addition, LBSNs provide "location-specific data", in which users may check in at nearly same geographical coordinates but at very different venues. For example, users can check in at a cinema or a restaurant of the same shopping mall where both venues share the same geographical coordinates. In contrast, cell phone data provides coarse location accuracy and cannot differentiate between users' presences across different floors in the same building. The active partic-

[1] http://wikitravel.org/
[2] http://www.lonelyplanet.com/
[3] http://www.yoursingapore.com/
[4] http://www.australiatravel.com/
[5] A check-in is a user's status message in a LBSN with the purpose of letting his/her friends and/or public know his/her current location.

Table I. User-Venue Matrix (Values indicate number of visits.)

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}
u_1	0	0	0	0	0	10	5	3	0	1
u_2	0	0	0	0	0	21	15	3	0	12
u_3	0	0	0	0	0	1	0	3	3	1
u_4	0	0	0	0	0	10	5	0	4	3
u_5	1	11	5	3	1	0	0	0	0	0
u_6	3	9	0	3	2	0	0	0	0	0
u_7	7	1	1	0	1	0	0	0	0	0
u_8	2	4	3	3	4	0	0	0	0	0

Table II. "?" stands for preferences to be predicted.

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}
u_1	?	?	?	?	?	10	5	3	0	1
u_2	?	?	?	?	?	21	15	3	0	12
u_3	?	?	?	?	?	1	0	3	3	1
u_4	?	?	?	?	?	10	5	0	4	3
u_5	1	11	5	3	1	?	?	?	?	?
u_6	3	9	0	3	2	?	?	?	?	?
u_7	7	1	1	0	1	?	?	?	?	?
u_8	2	4	3	3	4	?	?	?	?	?

ipation of Foursquare users and the fine-grained venue annotations make personalized recommendation of locally interesting venues possible.

Collaborative filtering (CF) based approaches [Goldberg et al. 1992; Herlocker et al. 1999] seem to be plausible solutions to this problem demonstrated by their great successes in commercial applications, such as Amazon [Linden et al. 2003], Netflix [Bell and Koren 2007], Tivo [Ali and van Stam 2004], eBay [Yuan et al. 2011] and research on point-of-interest (POI) recommendations [Ye et al. 2011; Cheng et al. 2012; Zhou et al. 2012; Ying et al. 2012]. These approaches, including user-based CF and item-based CF, automatically generate recommended items of a user using known preferences of other users or known preferences of other items. However, CF-based algorithms, being memory or model based, require sufficient overlaps among users in terms of items rated so that the correspondences among users or items can be readily identified. In LBSNs, however, users usually visit venues that are within short geographical distances apart from their homes [Cheng et al. 2011a; Cho et al. 2011]. It is thus hard if not impossible to correlate users if they visit very different sets of venues with little/no overlaps. Let's consider the user-venue matrix shown in Table I where users $\{u_1, \cdots, u_4\}$ never visited venues $\{v_1, \cdots, v_5\}$ and users $\{u_5, \cdots, u_8\}$ never visited venues $\{v_6, \cdots, v_{10}\}$. If we were to use traditional CF techniques, the ratings marked with "?" in Table II would be hard to be estimated. In addition, most CF algorithms are static models in which relations are assumed to be fixed at different times. However, users' visiting behaviours often evolve over time [Noulas et al. 2011] and exhibit strong temporal patterns, such as daily/weekly patterns and demonstrate periodic property [Cheng et al. 2011a]. For example, people perform more check-ins at restaurants during meal times and visit shops mostly during weekends and weekday evenings. Hence, it also requires an effective way to incorporate the temporal information.

In this work, we aim to provide tourists with personalized location recommendations leveraging rich user generated location contents from LBSNs. Specifically, we identify *locally interesting* venues to be those frequently visited by local people but obscure to most foreigners. We make use of these *digital footprints* [Girardin et al. 2008] to understand collective local user behaviours and then provide venue recommendations to tourists from a global understanding via cross region communities' matching. Fig. 2

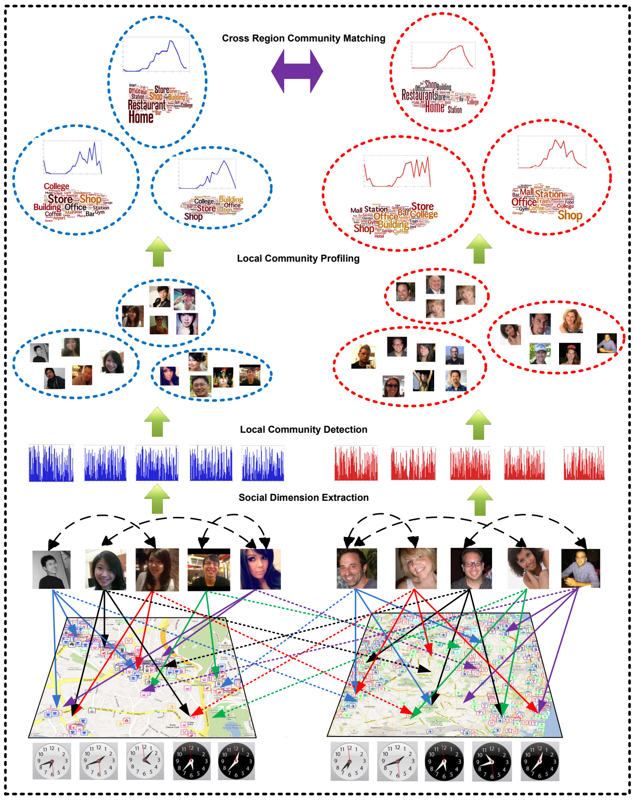


Fig. 2. Overall framework of locally interesting venue recommendations to tourists. It includes four components: 1. Social dimensions extraction (Section 3), 2. Local interest communities detection (Section 4.1), 3. Community profiling and representation (Section 4.1) and 4. Recommendations generation via cross region community matching (Section 4.2). Venues of same color are similar venues and dotted arrows indicate users' foreign visits to be recommended.

shows the overall framework which consists of four components. To tackle the sparseness problem and handle time-dependent varied behaviours, we propose to first extract users' latent *social dimensions* [Tang and Liu 2009] to capture users' preferences according to their local check-ins at different times, social relations and similarities among the visited venues. Social dimensions reflect users' latent drivers of their social behaviours and each dimension represents a plausible interest community among users. To accomplish this subtask, we propose a novel framework named Bayesian probabilistic tensor factorization with social and location regularization (BPTFSLR) that puts users' visiting behaviours, social relations and venue similarities into a unified framework. We next mine local interest communities in each geographical region using adaptive affinity propagation. We then represent each local community using global properties, such as venue categories and time of visits according to aggregated behaviours of community members. Finally, we correlate communities at different geographical regions to generate personalized recommendations of locally interesting venues to tourists. By conducting experiments on a representative real-world dataset, we demonstrate that our proposed scheme is effective in generating personalized recommendations of locally interesting venues to tourists.

With the effective extraction of users' latent social dimensions, further applications such as targeted advertising, content placement and caching and more relevant information diffusion can be built upon with more accurate preference matches.

The main contributions of this work are summarized as follows.

(1) To the best of our knowledge, this is the first work that targets at personalized recommendations of locally interesting venues to tourists.
(2) We propose a novel unified framework to effectively extract users' latent social dimensions. The framework considers users' temporal visiting behaviours, social ties and venue similarities simultaneously.
(3) We develop a novel idea to match users' local preferences across geographical regions using aggregated behaviours via global properties.

The rest of the paper is organized as follows. Section 2 formally defines the problem. Section 3 and 4 detail the social dimensions extraction and the recommendations generation, respectively. Section 5 reports the experimental results. Section 6 reviews the related work. Finally, Section 7 gives the conclusions and future work.

2. PROBLEM DEFINITION

In this section, we formally define the problem statement. It is worth mentioning that the problem we study and the method we propose in this paper are applicable to all LBSNs and we choose Foursquare as the testbed in this work.

Problem Statement: Let $\mathbb{U}^g = \{u_1^g, \cdots, u_{N_g}^g\}$ be a set of users and $\mathbb{V}^g = \{v_1^g, \cdots, v_{M_g}^g\}$ be a set of venues in geographical region g. Let $\mathbb{T} = \{t_1, \cdots, t_T\}$ be a set of location-independent time periods. We define a set of check-ins $\mathbb{C}^g = \{c_1^g, \cdots, c_{q_g}^g\}$, where each check-in is a tuple: (u_i^g, v_j^g, t_k) indicating that user u_i^g visits venue v_j^g at time t_k in region g. Let $\mathbb{G}^g = (\mathbb{U}^g, \mathbb{E}_1^g)$ be the undirected social network graph in region g, where $\mathbb{E}_1^g$ represents the social relations between users in region g. We then define the corresponding adjacency matrix $\mathbf{R}^g \in \mathbb{R}^{N\times N}$, where R_{ri}^g is the strength of the social relation between user r and i in region g. Let $\mathbb{H}^g = (\mathbb{V}^g, \mathbb{E}_2^g)$ be the undirected venue relation graph in region g. We next define the corresponding adjacency matrix $\mathbf{B}^g \in \mathbb{R}^{M\times M}$, where B_{jl}^g represents venue similarity between venue j and l in region g. Given $\mathbb{C}^g$, $\mathbb{G}^g$, $\mathbb{H}^g$ and $\mathbb{T}$, where $g = 1, 2, \cdots$, our aim is to recommend a list of locally interesting venues $\{v_1^a, \cdots, v_{L^a}^a\}$ in region a to users $\{u_1^b, \cdots, u_{N_b}^b\}$ from region b when they visit region a, where a is geographically different from b, L^a is the number of locally interesting venues in region a and N_b is the number of users in region b.

3. SOCIAL DIMENSIONS EXTRACTION

In LBSNs, users exhibit heterogenous visiting behaviours, which naturally classify them into different interest groups, such as food lovers, shoppers, etc. In addition, even within the same interest groups, people may exhibit further different preferences. For example, sports lovers may have different exercising preferences in terms of venues and times: some prefer jogging in the morning in their neighbourhoods; some like to exercise during weekends in nature parks; and some others may prefer to exercise in gyms after work. The inherent heterogenous user preferences make it hard to interpret the connections between people in social networks. Towards gaining insights on the underlying users' interests, [Tang and Liu 2009] formally defined social dimensions of each user with each dimension representing a latent affiliation among users in order to approximate direct different connections. In this section, we present a unified framework for effective extraction of latent social dimensions for each user by simultaneously considering temporal factors and various relations among different entities.

3.1. Matrix Factorization Model

A simple approach to extract the latent social dimensions is to use probabilistic matrix factorization (PMF) [Salakhutdinov and Mnih 2008b], where the underlying assumption is that both users and venues can be modeled by a set of latent representations. Let $\mathbf{Q} \in \mathbb{R}^{N \times M}$ be the $user \times venue$ matrix, where Q_{ij} is the preference of user i towards venue j and is computed based on the number of times i visits j as follows.

$$Q_{ij} = \frac{c(i,j)}{\sum_{j'=1}^{M} c(i,j')} , \tag{1}$$

where $c(i,j)$ is the number of times user i checks in at venue j. Let $\mathbf{U} \in \mathbb{R}^{D \times N}$ be the collection of latent social dimensions of users with each column $\mathbf{u}_i$ representing a D-dimensional latent social dimensions for user i and $\mathbf{V} \in \mathbb{R}^{D \times M}$ be the latent venue feature matrix with each column $\mathbf{v}_j$ representing a D-dimensional latent feature vector for venue j. PMF then approximates Q_{ij} based on the inner-product of corresponding latent features, i.e. $Q_{ij} \approx \mathbf{u}_i^T \mathbf{v}_j$. The conditional probability of the observed preferences is defined as:

$$p(\mathbf{Q}|\mathbf{U},\mathbf{V}) = \prod_{i=1}^{N} \prod_{j=1}^{M} \left[\mathcal{N}(Q_{ij}|\mathbf{u}_i^T \mathbf{v}_j, \tau_Q^{-1}) \right]^{I_{ij}} , \tag{2}$$

where $\mathcal{N}(\cdot|\mu,\tau^{-1})$ denotes the Gaussian distribution with mean μ and precision τ. Here I_{ij} is the indicator function that equals to 1 if user i ever visits venue j and 0 otherwise. In addition, zero-mean spherical Gaussian priors are imposed on $\mathbf{u}_i$ and $\mathbf{v}_j$ to control the model complexity: $\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \sigma_U^2 \mathbf{I})$ for $i = 1, \cdots, N$ and $\mathbf{v}_j \sim \mathcal{N}(\mathbf{0}, \sigma_V^2 \mathbf{I})$ for $j = 1, \cdots, M$.

Assuming $\mathbf{U}$ and $\mathbf{V}$ are independently distributed, we can maximize the log-posterior over $\mathbf{U}$ and $\mathbf{V}$ as follows.

$$\mathbf{U}^*, \mathbf{V}^* = \arg\max_{\mathbf{U},\mathbf{V}} p(\mathbf{U},\mathbf{V}|\mathbf{Q}) = \arg\max_{\mathbf{U},\mathbf{V}} p(\mathbf{U})p(\mathbf{V})p(\mathbf{Q}|\mathbf{U},\mathbf{V}). \tag{3}$$

It turns out that the learning procedure corresponds to the following weighted regularized matrix factorization:

$$\mathbf{U}^*, \mathbf{V}^* = \arg\min_{\mathbf{U},\mathbf{V}} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} (Q_{ij} - \mathbf{u}_i^T \mathbf{v}_j)^2 + \frac{\lambda_U}{2} \sum_{i=1}^{N} \|\mathbf{u}_i\|_2^2 + \frac{\lambda_V}{2} \sum_{j=1}^{M} \|\mathbf{v}_j\|_2^2 , \tag{4}$$

where $\lambda_U = (\tau_Q \sigma_U^2)^{-1}$ and $\lambda_V = (\tau_Q \sigma_V^2)^{-1}$. The local minimum of this non-convex optimization problem can be efficiently found by stochastic gradient descent [Bottou 2004]. Alternatively, to avoid parameter tuning and achieve automatic control of model complexity, we can also apply a full Bayesian treatment using markov chain monte carlo (MCMC) to obtain the posterior probability distribution of the user latent social dimensions [Salakhutdinov and Mnih 2008a]. However, PMF does not consider the temporal factors and assumes consistent users' behaviour across different time periods.

3.2. Tensor Factorization Model

The previous approach assumes that visiting preferences are fixed at different times. However, time factors are strong drives which inherently direct users' movements and users' visiting behaviours exhibit significantly different temporal patterns in the real world [Eagle and Pentland 2009; Cheng et al. 2011a]. The visiting preferences are affected by two temporal aspects. First, users visit different venues at different times of the day. For example, people often visit food courts or restaurants during meal times and watch movies during the evening on Friday and weekends. Second, users exhibit different lifestyles in weekdays and weekends. [Noulas et al. 2011] reported drastic differences among types of venues visited at weekdays and weekends. For example, more people check in at cafes and offices during weekday mornings while an increasing number of check-ins are registered at bars during weekday evenings and weekend afternoons. In addition, venues such as offices are rarely visited during weekends whereas leisure venues such as hotels receive a significant number of visits during weekends. To bring in the time factors, we employ probabilistic tensor factorization (PTF) to model the time-evolving preferences [Xiong et al. 2010]. With the introduction of time factors, the $user \times venue$ two-dimensional matrix is extended to the $user \times venue \times time$ three-dimensional tensor. We consider splitting users' visiting times into eight periods: $\{morning\ (5am - 11am),\ afternoon\ (12pm - 18pm),\ evening\ (19pm - 23pm),\ night\ (12am - 4am)\} \times \{weekday,\ weekend\}$.

Extended from the relational data in matrix factorization model, let $\boldsymbol{Q} \in \mathbb{R}^{N \times M \times T}$ be the $user \times venue \times time$ tensor, where Q_{ij}^k is the preference of user i towards venue j at time k and can be computed based on the number of times i visits j at k as follows.

$$Q_{ij}^k = \frac{c^k(i,j)}{\sum_{j'=1}^{M} c^k(i,j')},\ k = 1, 2, \cdots, T, \tag{5}$$

where $c^k(i,j)$ is the number of times user i visits venue j at time k. Extending the idea of PMF, we can approximate Q_{ij}^k with the inner-product of three D-dimensional vectors:

$$Q_{ij}^k \approx \langle \mathbf{u}_i, \mathbf{v}_j, \mathbf{t}_k \rangle = \sum_{d=1}^{D} U_{di} V_{dj} T_{dk}, \tag{6}$$

where $\mathbf{t}_k$ is the additional latent feature vector for the kth time factor. Intuitively, Eq (6) makes the visiting preferences not only depend on how similar a user's preferences and a venue's preferences are, but also on how much these preferences match with the current crowd behaviours which are reflected by the time factors. We then extend the conditional probability of the observed preferences as:

$$p(\mathbf{Q}|\mathbf{U}, \mathbf{V}, \mathbf{T}) = \prod_{i=1}^{N} \prod_{j=1}^{M} \prod_{k=1}^{T} \left[\mathcal{N}(Q_{ij}^k | \langle \mathbf{u}_i, \mathbf{v}_j, \mathbf{t}_k \rangle), \tau_Q^{-1} \right]^{I_{ij}^k}. \tag{7}$$

To avoid overfitting, similarly, we impose zero-mean, independent Gaussian priors on user and venue latent vectors as before. Following [Xiong et al. 2010], we assume that the time factors change smoothly over time and depend only on their immediate predecessor where we also assume that the Markov property holds. Thus, the conditional prior for $\mathbf{T}$ and the initial time feature vector $\mathbf{t}_0$ are defined as:

$$P(\mathbf{t}_k) = \mathcal{N}(\mathbf{t}_{k-1}, \sigma_T^2 \mathbf{I}), \quad P(\mathbf{t}_0) = \mathcal{N}(\boldsymbol{\mu}_T, \sigma_0^2 \mathbf{I}). \tag{8}$$

We can maximize the log-posterior over $\mathbf{U}, \mathbf{V}, \mathbf{T}$ as follows.

$$\mathbf{U}^*, \mathbf{V}^*, \mathbf{T}^* = \arg\max_{\mathbf{U},\mathbf{V},\mathbf{T}} p(\mathbf{U}, \mathbf{V}, \mathbf{T}|\mathbf{Q}) = \arg\max_{\mathbf{U},\mathbf{V},\mathbf{T}} p(\mathbf{U}, \mathbf{V}, \mathbf{T}) p(\mathbf{Q}|\mathbf{U}, \mathbf{V}, \mathbf{T}). \tag{9}$$

With the independence assumption, after mathematical derivations, the optimization problem becomes:

$$\mathbf{U}^*, \mathbf{V}^*, \mathbf{T}^* = \arg\min_{\mathbf{U},\mathbf{V},\mathbf{T}} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{k=1}^{T} \left[Q_{ij}^k - \langle \mathbf{u}_i, \mathbf{v}_j, \mathbf{t}_k \rangle \right]^2 + \frac{\lambda_U}{2} \sum_{i=1}^{N} \|\mathbf{u}_i\|_2^2 + \frac{\lambda_V}{2} \sum_{j=1}^{M} \|\mathbf{v}_j\|_2^2 + \frac{\lambda_T}{2} \sum_{k=1}^{T} \|\mathbf{t}_k - \mathbf{t}_{k-1}\|_2^2 + \frac{\lambda_0}{2} \|\mathbf{t}_0 - \boldsymbol{\mu}_T\|_2^2, \tag{10}$$

where $\lambda_U = (\tau_Q \sigma_U^2)^{-1}$, $\lambda_V = (\tau_Q \sigma_V^2)^{-1}$, $\lambda_T = (\tau_Q \sigma_T^2)^{-1}$ and $\lambda_0 = (\tau_Q \sigma_0^2)^{-1}$. We can adopt the same stochastic gradient descent approach to find local minimums of this non-convex optimization problem. Similarly, we can also adopt the Bayesian treatment and use MCMC methods to obtain the posterior distribution of users' latent social dimensions [Xiong et al. 2010]. However, PTF does not take users' social relations and venue similarities into consideration.

3.3. Regularized Tensor Factorization

The formulation in Section 3.2 has considered the temporal variations of users' visiting behaviours. In this section, we further extend the previous formulation by simultaneously considering the social ties and inter-venue similarities in LBSNs in order to achieve more accurate extraction of users' social dimensions.

3.3.1. Social Relation. Intuitively, "friends" tend to have similar behaviours and preferences. For example, a group of friends may often visit the same restaurants for gathering or hang out to watch movies together. A user may also visit certain places which are recommended by his/her friends. These suggest that it is useful to consider social ties to bring "friends" closer to each other in the latent space. Following [Ye et al. 2011], we consider two factors to relate users in LBSNs. First, friends who have more common friends may have better trust in terms of their recommendations, thus we consider the overlapping levels of their friend sets. Second, friends sharing more check-ins should have more similar tastes, thus we consider the overlapping levels of their check-in sets.

We define the user similarity as follows. Given the user set $\mathbb{U} = \{u_1, \cdots, u_N\}$, their friend set $\{\mathbb{F}_1, \cdots, \mathbb{F}_N\}$ and their check-in set $\{\mathbb{V}_1, \cdots, \mathbb{V}_N\}$, we introduce $\alpha \in [0, 1]$ as a tuning parameter and define the user similarity matrix $\mathbf{R} \in \mathbb{R}^{N \times N}$, where R_{ri} is computed as follows.

$$R_{ri} = \begin{cases} \alpha \cdot \frac{|\mathbb{F}_r \cap \mathbb{F}_i|}{|\mathbb{F}_r \cup \mathbb{F}_i|} + (1-\alpha) \cdot \frac{|\mathbb{V}_r \cap \mathbb{V}_i|}{|\mathbb{V}_r \cup \mathbb{V}_i|} & \text{if } u_r \in \mathbb{F}_i, \\ 0 & \text{Otherwise.} \end{cases} \tag{11}$$

3.3.2. Venue Similarity. Venues have different social functions. In addition to categories, venues are also enriched with users' comments about the activities, reviews and descriptions. In Foursquare, users are free to write tips, which may cover a variety of diverse topics at venues. For example, a tip left at an art museum may recommend a special exhibition or give positive/negative comments on the museum environment. We argue that tips sometimes provide better evidences than categories to describe venues. For example, during the examination reading weeks, venues such as libraries, school canteens, study rooms and Starbucks in universities, though belong to different categories, and tend to have similar social functions: places for preparing exams. We thus seek to model venue similarities using the associated tips.

We aggregate all tips of a venue and perform the below steps to filter the noise and reduce the feature space:

— We tokenize text descriptions and put them into lowercase.
— We remove all the non-alphanumeric characters.

— We remove rare terms (terms with frequency< 5).

Then, the text descriptions for each venue v_j are represented as a word-frequency vector $\mathbf{w}_j = [w_j(1) \cdots w_j(Z)]$, where $w_j(b)$ denotes the frequency of term b in the text descriptions of venue v_j and Z is the vocabulary size. We then define the venue similarity matrix $\mathbf{B} \in \mathbb{R}^{M \times M}$, where $B_{jl} = \frac{\mathbf{w}_j \cdot \mathbf{w}_l}{|\mathbf{w}_j| \cdot |\mathbf{w}_l|}$.

3.3.3. The Complete Formulation. With the introduction of user relations and venue similarities, we now present the complete formulation. Let $\mathbf{S} \in \mathbb{R}^{D \times N}$ be the auxiliary user factor feature matrix and $\mathbf{D} \in \mathbb{R}^{D \times M}$ be the auxiliary venue factor feature matrix. We have the conditional distribution of user and venue similarities as follows.

$$p(\mathbf{R}|\mathbf{S}, \mathbf{U}) = \prod_{r=1}^{N} \prod_{i=1}^{N} \left[\mathcal{N}(R_{ri}|\mathbf{s}_r^T \mathbf{u}_i, \tau_R^{-1})\right]^{I_{ri}^R}, \tag{12}$$

$$p(\mathbf{B}|\mathbf{V}, \mathbf{D}) = \prod_{j=1}^{M} \prod_{l=1}^{M} \left[\mathcal{N}(B_{jl}|\mathbf{v}_j^T \mathbf{d}_l, \tau_B^{-1})\right]^{I_{jl}^B}. \tag{13}$$

As before, we introduce zero-mean, independent Gaussian priors on the two introduced feature matrices. Assuming user similarities, venue similarities and user visiting preferences are independently distributed conditioned on the latent factors, we may estimate $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}$ by maximizing the logarithm of the posterior distribution of the observed similarities and preferences:

$$\begin{aligned}
&\mathbf{U}^*, \mathbf{V}^*, \mathbf{S}^*, \mathbf{D}^*, \mathbf{T}^* \\
&= \arg \max_{\mathbf{U},\mathbf{V},\mathbf{S},\mathbf{D},\mathbf{T}} p(\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}|\mathbf{R}, \mathbf{B}, \mathbf{Q}) \\
&= \arg \max_{\mathbf{U},\mathbf{V},\mathbf{S},\mathbf{D},\mathbf{T}} p(\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}) p(\mathbf{R}, \mathbf{B}, \mathbf{Q}|\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}).
\end{aligned} \tag{14}$$

Maximizing the log posterior with respect to $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}$ and $\mathbf{T}$ is equivalent to minimizing the following objective function with quadratic regularization terms:

$$\begin{aligned}
\boldsymbol{L}(\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}) &= \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{k=1}^{T} I_{ij}^k [Q_{ij}^k - \langle \mathbf{u}_i, \mathbf{v}_j, \mathbf{t}_k \rangle]^2 \\
&+ \frac{\lambda_R}{2} \sum_{r=1}^{N} \sum_{i=1}^{N} I_{ri}^R [R_{ri} - \mathbf{s}_r^T \mathbf{u}^i]^2 \\
&+ \frac{\lambda_B}{2} \sum_{j=1}^{M} \sum_{l=1}^{M} I_{jl}^B [B_{jl} - \mathbf{v}_j^T \mathbf{d}^l]^2 \\
&+ \frac{\lambda_D}{2} \sum_{l=1}^{M} \|\mathbf{d}_l\|_2^2 + \frac{\lambda_T}{2} \sum_{k=1}^{T} \|\mathbf{t}_k - \mathbf{t}_{k-1}\|_2^2 + \frac{\lambda_0}{2} \|\mathbf{t}_0 - \boldsymbol{\mu}_T\|_2^2 \\
&+ \frac{\lambda_U}{2} \sum_{i=1}^{N} \|\mathbf{u}_i\|_2^2 + \frac{\lambda_S}{2} \sum_{r=1}^{N} \|\mathbf{s}_r\|_2^2 + \frac{\lambda_V}{2} \sum_{j=1}^{M} \|\mathbf{v}_j\|_2^2,
\end{aligned} \tag{15}$$

where $\lambda_R = \frac{\tau_R}{\tau_Q}$, $\lambda_B = \frac{\tau_B}{\tau_Q}$, $\lambda_D = (\tau_Q \sigma_D^2)^{-1}$, $\lambda_T = (\tau_Q \sigma_T^2)^{-1}$, $\lambda_0 = (\tau_Q \sigma_0^2)^{-1}$, $\lambda_U = (\tau_Q \sigma_U^2)^{-1}$, $\lambda_S = (\tau_Q \sigma_S^2)^{-1}$ and $\lambda_V = (\tau_Q \sigma_V^2)^{-1}$.

The objective function is non-convex, and we may only be able to find a local minimum by iteratively updating the latent feature vectors using methods such as the stochastic gradient descent. One issue with this approach is parameter-tuning. Since there are eight of them, the usual approach of parameter selection, such as cross-validation is infeasible even for a modest problem size. Here, in the spirit of [Xiong et al. 2010], we seek a full Bayesian treatment to average out the hyperparameters in the model, which both helps to alleviate overfitting and saves us from the painful parameter tuning.

3.3.4. Learning By Markov Chain Monte Carlo. The full Bayesian treatment integrates out all model parameters and hyperparameters and arrives at a predictive distribution of future observations given previous observed data. Since this predictive distribution is obtained by averaging all models in the model space specified by the priors, it is less likely to overfit the given set of observations. However, when integrating over parameters one often cannot obtain an analytical solution, thus we resort to sampling-based approximation methods, in particular, MCMC [Andrieu et al. 2003].

To generate user similarity R_{ri}, venue similarity B_{jl} and user visiting preference Q_{ij}^k, we first sample $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}$ according to Eq (17), and then sample $\mathbf{T}$ according to Eq (18). R_{ri}, B_{jl} and Q_{ij}^k can then be generated according to Eq (12), Eq (13) and Eq (7) respectively. Fig. 3 shows the graphical model of the entire generative process.

The key ingredient of the fully Bayesian treatment is to view the hyperparameters: τ_Q, τ_R, τ_B, $\Theta_U \equiv \{\boldsymbol{\mu}_U, \boldsymbol{\Lambda}_U\}$, $\Theta_V \equiv \{\boldsymbol{\mu}_V, \boldsymbol{\Lambda}_V\}$, $\Theta_S \equiv \{\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S\}$, $\Theta_D \equiv \{\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D\}$ and $\Theta_T \equiv \{\boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T\}$ as random variables as showed in Fig. 3. We choose the prior distributions for the hyperparameters as follows.

$$
\begin{aligned}
&p(\tau_Q) = \mathcal{W}(\tau_Q|W_1, v_0), \;\; p(\tau_R) = \mathcal{W}(\tau_R|W_1, v_0), \;\; p(\tau_B) = \mathcal{W}(\tau_B|W_1, v_0),\\
&p(\Theta_U) = p(\boldsymbol{\mu}_U|\boldsymbol{\Lambda}_U)p(\boldsymbol{\Lambda}_\mathbf{U}) = \mathcal{N}(\boldsymbol{\mu}_0, (\beta\boldsymbol{\Lambda}_U)^{-1})\mathcal{W}(\Lambda_U|\mathbf{W}_0, v_0),\\
&p(\Theta_V) = p(\boldsymbol{\mu}_V|\boldsymbol{\Lambda}_V)p(\boldsymbol{\Lambda}_\mathbf{V}) = \mathcal{N}(\boldsymbol{\mu}_0, (\beta\boldsymbol{\Lambda}_V)^{-1})\mathcal{W}(\Lambda_V|\mathbf{W}_0, v_0),\\
&p(\Theta_S) = p(\boldsymbol{\mu}_S|\boldsymbol{\Lambda}_S)p(\boldsymbol{\Lambda}_\mathbf{S}) = \mathcal{N}(\boldsymbol{\mu}_0, (\beta\boldsymbol{\Lambda}_S)^{-1})\mathcal{W}(\Lambda_S|\mathbf{W}_0, v_0),\\
&p(\Theta_D) = p(\boldsymbol{\mu}_D|\boldsymbol{\Lambda}_D)p(\boldsymbol{\Lambda}_\mathbf{D}) = \mathcal{N}(\boldsymbol{\mu}_0, (\beta\boldsymbol{\Lambda}_D)^{-1})\mathcal{W}(\Lambda_D|\mathbf{W}_0, v_0),\\
&p(\Theta_T) = p(\boldsymbol{\mu}_T|\boldsymbol{\Lambda}_T)p(\boldsymbol{\Lambda}_\mathbf{T}) = \mathcal{N}(\boldsymbol{\mu}_1, (\beta\boldsymbol{\Lambda}_T)^{-1})\mathcal{W}(\Lambda_T|\mathbf{W}_0, v_0),
\end{aligned}
\tag{16}
$$

where $\mathcal{W}(\cdot|\mathbf{V}, n)$ is a Wishart distribution of a $D \times D$ random matrix with n degrees of freedom and a scale matrix $\mathbf{V} \in \mathbb{R}^{D\times D}$. The hyperpriors are: $\mathbf{W}_0$, W_1, β, v_0, v_1, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$, which reflect the prior knowledge about the specific problem. In the Bayesian paradigm, they have little impact on the final predictions as reported in [Andrieu et al. 2003]. Next, the prior distributions for $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}$ are assumed to be Gaussian as before. However, the mean and the precision matrix may take arbitrary values:

$$
\begin{aligned}
p(\mathbf{u}_i) &= \mathcal{N}(\boldsymbol{\mu}_U, \boldsymbol{\Lambda}_U^{-1}), i = 1, \cdots, N,\\
p(\mathbf{v}_j) &= \mathcal{N}(\boldsymbol{\mu}_V, \boldsymbol{\Lambda}_V^{-1}), j = 1, \cdots, M,\\
p(\mathbf{s}_r) &= \mathcal{N}(\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S^{-1}), r = 1, \cdots, N,\\
p(\mathbf{d}_l) &= \mathcal{N}(\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D^{-1}), l = 1, \cdots, M.
\end{aligned}
\tag{17}
$$

For the time feature vectors, we make the same Markov assumption and consider the priors:

$$
p(\mathbf{t}_k) = \mathcal{N}(\mathbf{t}_{k-1}, \boldsymbol{\Lambda}_T^{-1}), k = 2, \cdots, T, \;\; p(\mathbf{t}_1) = \mathcal{N}(\boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T^{-1}). \tag{18}
$$

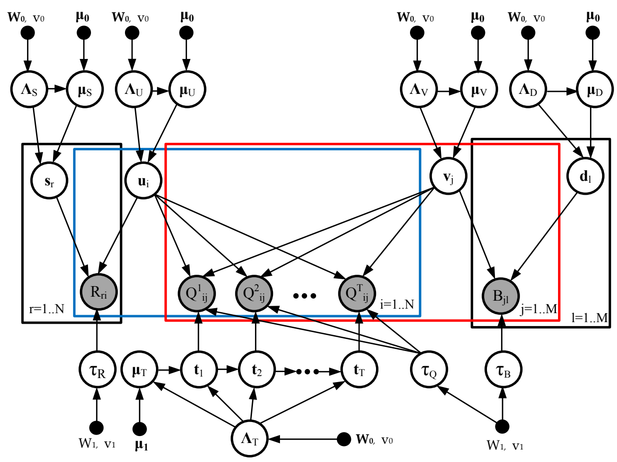


Fig. 3. The graphical model of probabilistic tensor factorization with user regularization: $\mathbf{R}$ and location regularization: $\mathbf{B}$ (BPTFSLR).

There are a few different classes of MCMC. Here we adopt Gibbs sampling [Gelfand and Smith 1990], where the target random variables: $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}$ are decomposed into several blocks and at each iteration a block of random variables is sampled while all the others are fixed until the process converges. The outline of the sampling procedure is shown in Algorithm 1 where the explicit updated conditional distributions of hyperparameters and model parameters are described in the online appendix.

4. RECOMMENDATIONS GENERATION

The extracted latent social dimensions of each user are expected to reveal an underlying partition of local interest groups at each region. Local interest communities may exhibit unique local behaviours. For example, some users in Singapore may often visit Crystal Jade - La Mian Xiao Long Bao while some users in New York City may frequently go to Congee Village. While, at the global level, these very different communities are expected to correlate well with each other: In the previous example, people in the two groups belong to the same global interest group: Chinese food lovers. The correlations between communities then give clues on correlations between people across geographical regions. In this section, we describe how we make use of users' social dimensions to generate venue recommendations in a city other than users' home city by using cross region community matching.

4.1. Local Community Profiling

With the extracted users' underlying social dimensions, we seek to first group them according to their latent interests in the regional level [Zhao et al. 2013]. There are a number of approaches to detect communities or dense subgroups, such as clustering or community detection. However, we do not know the number of communities in a region beforehand. Also the number of interest communities may vary across different regions: There may be more communities in big cities where people exhibit more varied kinds of behaviours whereas there can be very few communities in smaller counties where people may have more homogeneous life patterns. To alleviate this problem, we resort to affinity propagation (AP), which operates by first simultaneously considering all users as potential community centres and then keep exchanging messages among them until a good set of communities emerges [Frey and Dueck 2007]. In this work, to

ALGORITHM 1: Gibbs sampling for BPTFSLR

Input: $\boldsymbol{Q}$: The user-venue-time tensor, $\mathbf{R}$: The social relation matrix, $\mathbf{B}$: The venue similarity matrix, n: The maximum number of iterations
Output: $\mathbf{U}$: The latent social dimensions
Initialize model parameters $\{\mathbf{U}^{(1)}, \mathbf{V}^{(1)}, \mathbf{S}^{(1)}, \mathbf{D}^{(1)}, \mathbf{T}^{(1)}\}$
for $a = 1$ *to* n **do**
$\tau_Q^{(a)} \sim p(\tau_Q|\mathbf{Q}, \mathbf{U}^{(a)}, \mathbf{V}^{(a)}, \mathbf{T}^{(a)})$ (Eq (22)), $\tau_R^{(a)} \sim p(\tau_R|\mathbf{R}, \mathbf{U}^{(a)}, \mathbf{S}^{(a)})$ (Eq (23)),
$\tau_B^{(a)} \sim p(\tau_B|\mathbf{B}, \mathbf{V}^{(a)}, \mathbf{D}^{(a)})$ (Eq (25)), $\Theta_U^{(a)} \sim p(\Theta_U|\mathbf{U}^{(a)})$ (Eq (27)),
$\Theta_V^{(a)} \sim p(\Theta_V|\mathbf{V}^{(a)})$ (Eq (29)), $\Theta_S^{(a)} \sim p(\Theta_S|\mathbf{S}^{(a)})$ (Eq (31)), $\Theta_D^{(a)} \sim p(\Theta_D|\mathbf{D}^{(a)})$ (Eq (33)),
$\Theta_T^{(a)} \sim p(\Theta_T^{(a)}|\mathbf{T}^{(a)})$ (Eq (34))
for $i = 1$ *to* N **do**
$\mathbf{u}_i^{(a+1)} \sim p(\mathbf{u}_i|\mathbf{Q}, \mathbf{V}^{(a)}, \mathbf{T}^{(a)}, \mathbf{R}, \mathbf{S}^{(a)}, \tau_Q^{(a)}, \tau_R^{(a)}, \Theta_U^{(a)})$ (Eq (37))
end
for $j = 1$ *to* M **do**
$\mathbf{v}_j^{(a+1)} \sim p(\mathbf{v}_j|\mathbf{Q}, \mathbf{U}^{(a+1)}, \mathbf{T}^{(a)}, \mathbf{B}, \mathbf{D}^{(a)}, \tau_Q^{(a)}, \tau_B^{(a)}, \Theta_V^{(a)})$ (Eq (38))
end
for $r = 1$ *to* N **do**
$\mathbf{s}_r^{(a+1)} \sim p(\mathbf{s}_r|\mathbf{R}, \mathbf{U}^{(a+1)}, \tau_R^{(a)}, \Theta_S^{(a)})$ (Eq (39))
end
for $l = 1$ *to* M **do**
$\mathbf{d}_l^{(a+1)} \sim p(\mathbf{d}_l|\mathbf{B}, \mathbf{V}^{(a+1)}, \tau_B^{(a)}, \Theta_D^{(a)})$ (Eq (40))
end
for $k = 1$ *to* T **do**
if $k = 1$ **then**
$\mathbf{t}_1^{(a+1)} \sim p(\mathbf{t}_1|\mathbf{Q}, \mathbf{U}^{(a+1)}, \mathbf{V}^{(a+1)}, \mathbf{t}_2^{(a)}, \tau_Q^{(a)}, \Theta_T^{(a)})$ (Eq (41))
else
if $k = 2, \cdots, T-1$ **then**
$\mathbf{t}_k^{(a+1)} \sim p(\mathbf{t}_k|\mathbf{Q}, \mathbf{U}^{(a+1)}, \mathbf{V}^{(a+1)}, \mathbf{t}_{k-1}^{(a+1)}, \mathbf{t}_{k+1}^{(a)}, \tau_Q^{(a)}, \Theta_T^{(a)})$ (Eq (41))
else
$\mathbf{t}_T^{(a+1)} \sim p(\mathbf{t}_T|\mathbf{Q}, \mathbf{U}^{(a+1)}, \mathbf{V}^{(a+1)}, \mathbf{t}_{T-1}^{(a+1)}, \tau_Q^{(a)}, \Theta_T^{(a)})$ (Eq (41))
end
end
end
end

avoid parameter tuning, we use adaptive affinity propagating (AAP) which improves AP by automatically adjusting the damping factor and preference during the learning process [Wang et al. 2008].

Given the set of interest communities detected in each geographical region, we aim to understand and represent each community by means of its group profiles [Zhao et al. 2013] so that the correspondences between communities at different regions can be readily created. According to the concept of Homophily [McPherson et al. 2001], connections occur at higher rate between similar people than dissimilar people, which makes it sensible to profile each group using attributes of its group members. We utilize two global properties related to the check-in behaviours. First, Foursquare provides a well-structured and hierarchically-organized venue categories[6]. Each Foursquare venue is mapped to one or more categories depending on its social function. Second, users visit venues at different times, which shows another dimen-

[6]http://aboutfoursquare.com/foursquare-categories/

sions related to users' behaviours. To utilize these two dimensions of information, we represent each community by a weighted vector, where each dimension represents a visit to a particular venue category at a particular time period. In total, we utilize 423 leaf categories and the eight different time periods, which are $\{morning\ (5am - 11am),\ afternoon\ (12pm - 18pm),\ evening\ (19pm\ -\ 23pm),\ night\ (12am\ -\ 4am)\} \times \{weekday,\ weekend\}$.

4.2. Venue Recommendations via Cross Region Community Matching

While it is possible to directly compare communities located in different regions using traditional vector comparison metrics, in this section, we seek a more robust community representation that is able to reduce the noise, which may be caused by possible occasionally irregular users' behaviours. Let $\mathbf{C}_a \in \mathbb{R}^{l\times k_a}$ and $\mathbf{C}_b \in \mathbb{R}^{l\times k_b}$ be the community representations at region a and b mentioned in section 4.1, respectively, where l is the dimension of community representation, k_a, k_b are the number of local interest communities in region a and b, respectively. The joint community representation of communities at these two regions are then $\mathbf{C}_{ab} = [\mathbf{C}_a\ \mathbf{C}_b] \in \mathbb{R}^{l\times(k_a+k_b)}$.

People usually have multiple interests with different strengths. For example, most of the tourists are interested in local food sampling and shopping but some of them are more interested in food while others prefer to spend more time on shopping. Thus, a community of people is inherently a mixture of users' interests with varying weights. Motivated by this, we seek to learn a set of p latent interest factors: $\mathbf{A} \in \mathbb{R}^{l\times p}$ and generate more robust community representations on top of these factors.

Sparse representation has been shown to be effective in noise reduction and data compression [Wright et al. 2009]. We thus adopt the non-negative matrix factorization with sparseness constraints proposed by [Hoyer 2004] to decompose the joint community representation into the latent interest factors $\mathbf{A}$ and the sparse community representations $\mathbf{X}$ by solving the following optimization problem:

$$\mathbf{A}, \mathbf{X} = \arg\min_{\mathbf{A},\mathbf{X}} \|\mathbf{C}_{ab} - \mathbf{AX}\|, \tag{19a}$$

$$\text{s.t.} \quad \text{sparseness}(\mathbf{a}_i) = s_a, \forall i, \tag{19b}$$

$$\text{sparseness}(\mathbf{x}_i) = s_x, \forall i, \tag{19c}$$

where $\mathbf{X} \in \mathbb{R}^{p\times(k_a+k_b)}$ is the sparse community representations, s_a and s_x are the desired sparseness of $\mathbf{A}$ and $\mathbf{X}$, respectively. Here sparseness$(\cdot)$ is the sparseness measure as defined in [Hoyer 2004].

Let user i belong to community $\mathbb{C}_T^a$ at region a, his/her predictive preference towards venue j at region b can then be computed as follows.

$$\hat{Q}_{ij} = \sum_k s(\mathbb{C}_T^a, \mathbb{C}_k^b) \sum_{i' \in \mathbb{C}_k^b} Q_{i'j}, \tag{20}$$

where $\mathbb{C}_k^b, k = 1, 2, \cdots$ are communities at region b and $s(\cdot,\cdot)$ is the cosine similarity between two communities' sparse representations.

5. EXPERIMENTS

In this section, we report the evaluation strategies and experimental results. We first evaluate the effectiveness of the latent social dimensions extraction in the traditional evaluation framework of collaborative filtering. We then report the performance on prediction of locally interesting venues to tourists using four regional subsets of the sampled Foursquare dataset. In addition, we want to verify the hypotheses below:

Table III. Properties of sampled popular regions: N is the number of active users, M_{L} is the number of local venues, C_{L} is the number of local check-ins, M_{F} is the number of foreign venues and C_{F} is the number of foreign check-ins.

	N	$M_{\rm L}$	$C_{\rm L}$	$M_{\rm F}$	$C_{\rm F}$
New York City	26,411	64,249	448,072	301,782	810,545
Chicago	7,138	36,164	353,290	120,940	341,651
Singapore	8,033	50,722	406,490	20,940	36,874
London	6,320	25,031	258,605	66,031	158,605

(1) The use of time factors help to improve venue prediction accuracy.
(2) The social and venue regularization leads to further improvement in the recommendation performance.
(3) Cross region community matching is able to generate relevant and accurate recommendation list for tourists.

5.1. Datasets

Usually, LBSNs such as Foursquare with rich location sensitive resources will restrict access of users' location data to their immediate social circles, such as "friends", and hence will not be available for public sampling. Instead, we turn to Twitter streams[7] where tweets containing check-in information are shared to the public. We monitor Twitter's streams and record each Foursquare check-in with keyword specified as "4sq". Each relevant tweet contains a short check-in message and a link pointing to the Foursquare check-in page, where we are able to retrieve more complete information related to the venues and users. In total, we have recorded $67,427,421$ check-ins performed by $1,067,818$ users at $3,923,267$ venues from August to November *2012*.

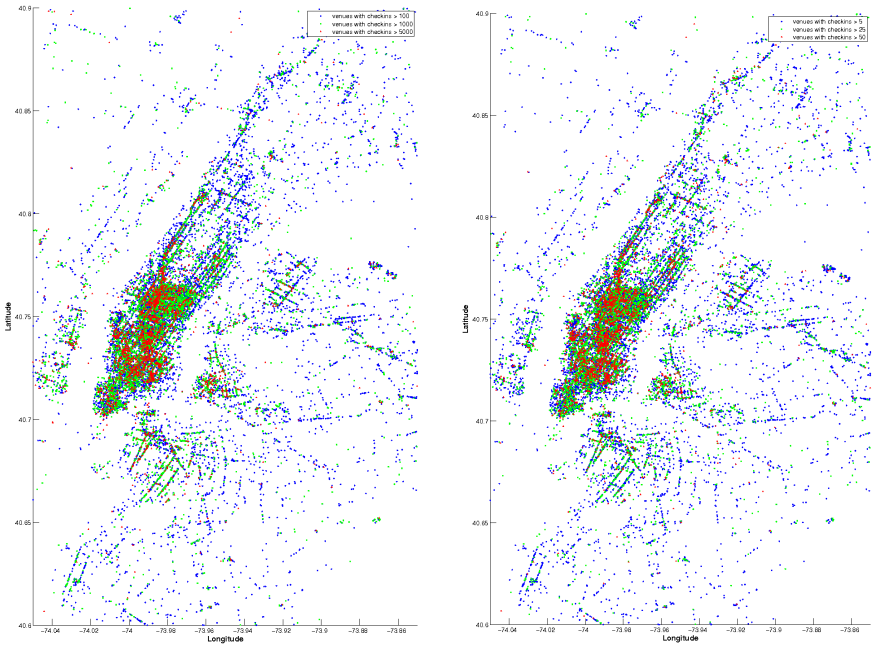
Not all check-ins are from genuine users. Motivated by the game elements such as badges, mayorships or free vouchers, some users may check in at certain venues with unrealistically high frequencies. In data preprocessing, we remove two kinds of suspicious check-ins. First, we remove check-ins from users who have performed more than ten check-ins within a minute. Second, we remove "sudden moves" where the two check-ins imply that a user is travelling at a speed faster than $1,000km/hour$ (Faster than the speed of normal commercial jet airplanes). In addition, we notice that certain venues are deleted by Foursquare in the housekeeping process. We remove all check-ins which were performed at these deleted venues.

Since our focus in this work is to generate locally interesting venue recommendations to tourists, we select users from Chicago (CHI), London (LDN), New York City (NYC) and Singapore (SG) and aim to recommend venues to them when they are in a city other than their home cities. We regard users' declared "homecity" in Foursquare as users' true home city. In addition, we remove a user if more than 50% of his/her check-ins are not in his/her declared home city. We then locate venues in different cities based on the geographical bounding boxes returned by Google's geocoding API[8]. Table III lists the properties of the four cities that we conduct experiments on. Active users are those who have performed at least ten check-ins during the crawling periods. Foreign venues for a user are check-ins performed at foreign venues.

5.2. Dataset Reliability/Representativeness

In this section, we aim to investigate whether the number of check-ins we obtained are reliable. This is because we sample the Foursquare check-ins by using Twitter streaming API, while not all users share their check-ins through Twitter. To verify that the

[7] https://dev.twitter.com/docs/streaming-api
[8] https://developers.google.com/maps/documentation/geocoding/



(a) Total check-ins (b) Sampled check-ins

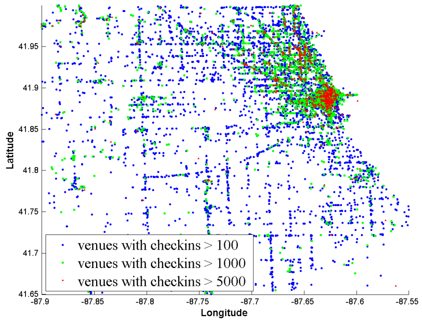
Fig. 4. Comparison between the popular venue distributions in New York City (JS Divergence = 0.3226)

check-ins we sampled are reliable, we count the number of check-ins for each venue of interest and compare it with its total number of check-ins reported by Foursquare. Let M be the number of venues of interest, we note that each check-in results in exactly one of M possible venues with probabilities $\mathbf{p} = (\frac{\sum_i c(i,1)}{\sum_{ij} c(i,j)}, \cdots, \frac{\sum_i c(i,M)}{\sum_{ij} c(i,j)})$. If we let the random variable h_j indicate the number of times venue j was visited over the n visits, then vector $\mathbf{h} = \{h_1, \cdots, h_M\}$ follows a multinomial distribution: $\mathbf{h} \sim \text{Multi}(n, \mathbf{p})$. We can then compute the similarity between the sampled distribution with the true distribution using Jensen–Shannon (JS) divergence. In addition, we visualize the venue distribution according to the number of check-ins in our sampled dataset and that of the total number of check-ins reported by Foursquare. Fig. (4, 5, 6 and 7) show that the sampled venues in New York City, Chicago, London and Singapore have similar relative popularity as compared to those reported by Foursquare.

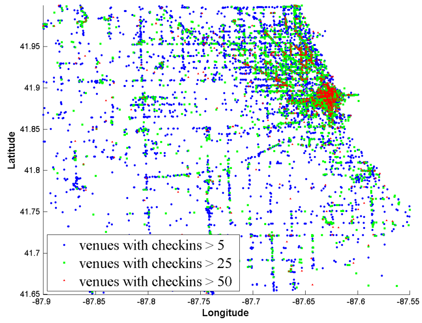
5.3. Parameter Settings

In this section, we describe the parameter settings. The parameters are tuned in another city: Los Angeles (LA). The number of active LA users is $6,389$, the number of local venues visited is $35,781$ and the number of local check-ins is $254,782$. We use the users' local visiting history from August to October *2012* as training set and use the check-ins performed during November *2012* as the testing set. The parameter settings are described as follows.

— User similarity tradeoff parameter: α. We tune α based on the prediction performance using the friend-based model. The predicted preference of user i towards venue j is computed as: $\hat{Q}_{ij} = \frac{\sum_{r \in \mathbb{U}, r \neq i} R_{ri} \cdot Q_{rj}}{\sum_{r \in \mathbb{U}, r \neq i} R_{ri}}$. We measure the performance using

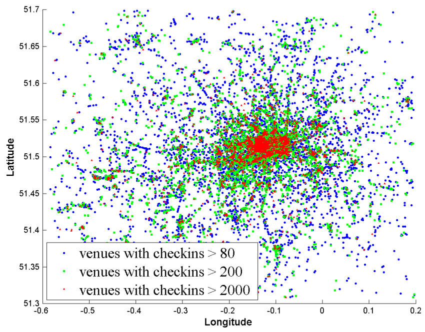


(a) Total check-ins

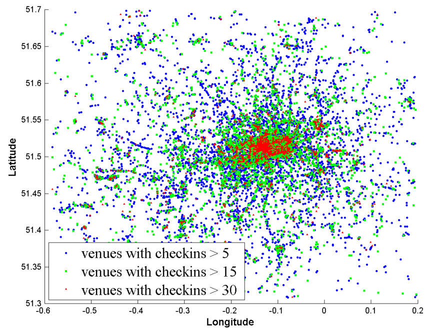


(b) Sampled check-ins

Fig. 5. Comparison between the popular venue distributions in Chicago (JS Divergence = 0.3351)

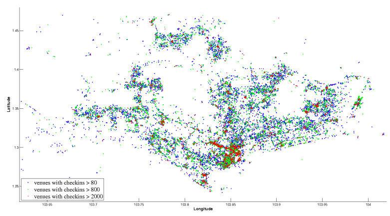


(a) Total check-ins

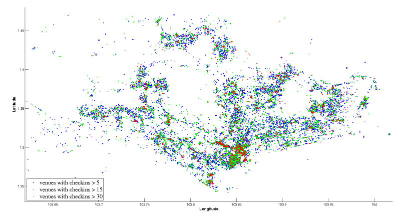


(b) Sampled check-ins

Fig. 6. Comparison between the popular venue distributions in London (JS Divergence = 0.4066)



(a) Total check-ins



(b) Sampled check-ins

Fig. 7. Comparison between the popular venue distributions in Singapore (JS Divergence = 0.2841)

MAP for $\alpha \in [0, 1]$ with increment: 0.05. The optimal performance is achieved when $\alpha = 0.35$.

— Parameters for Bayesian learning: $\mathbf{W}_0$, v_0, W_1, v_1, β, $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$. These parameters reflect our prior knowledge of the specific problem. Since Bayesian learning is able to adjust them according to the training data within a reasonably large range, we set them without tuning, similar to that of [Salakhutdinov and Mnih 2008a] and [Xiong et al. 2010]. The settings are: $\mathbf{W}_0 = \mathbf{I}$, $v_0 = D$, $W_1 = 1$, $v_1 = 1$, $\beta = 1$, $\boldsymbol{\mu}_0 = \mathbf{0}$ and $\boldsymbol{\mu}_1 = \mathbf{1}$, where $\mathbf{I}$ is a $D \times D$ identity matrix, $\mathbf{0}$ and $\mathbf{1}$ is a $D \times 1$ column vector of 0s and 1s, respectively.

— Number of samples used in Bayesian learning. Considering the tradeoff between the prediction accuracy and the computational cost, we empirically choose the number of samples to be 75 in this work.
— Latent dimension: D. We tune D based on the prediction performance using BPTF on LA users' local venue prediction. We determine $D = 60$ considering the tradeoff between the prediction accuracy and the computational cost.
— Parameters for AAP. We empirically set the parameters as follows. (1) Convergence condition: nconv $= 30$, (2) Initial damping factor: lam $= 0.5$ and (3) decreasing step of preferences: pstep $= 0.01$.
— Parameters for learning latent community representation: s_a, s_x and p. We empirically set $s_a = 0.5$ and $s_x = 0.85$ to allow moderate sparseness on the interest factors and high levels of sparseness on the community representations. And we empirically set $p = 200$.

5.4. Quality of Latent Social Dimensions

In this section, we investigate the quality of the users' latent social dimensions by our proposed BPTFSLR through evaluating the venue prediction accuracy in traditional collaborative filtering settings. We consider local venue predictions and aim to predict users' future visits based on their past visits. The details are as follows. For each city, we use users' check-ins performed from August to October *2012* as training set and check-ins performed during November *2012* as testing set. We remove all check-ins c_{ij} from the testing set for user i if there is a non-zero entry for c_{ij} in the training set. This filtering ensures that the prediction task is able to recover venues that the users haven't visited before. We then exclude users who do not share any check-in during the November period.

In addition, we further consider two kinds of settings. First, we generate general venue predictions without considering time factors. In this setting, methods that consider time information aggregate venue preferences at different time periods and provide a single ranking list with venues being sorted in the decreasing order of computed preferences. Second, we generate venue predictions for each user at different time periods. In this task, methods which do not consider time information are applied on the training and testing set for each time period individually and we report the averaged performance for all time periods. We name the first setting as "time-independent" setting and the second as "time-dependent" setting.

5.4.1. Evaluation Metrics. The aim of venue recommendation is to provide each user a ranking list of venues to visit. Thus, instead of predicting user-venue preferences, we aim to measure the quality of the ranking list of recommended venues against the ground truth. Similar to traditional information retrieval tasks, we use Precision@k, Recall@k and mean average precision (MAP) to report the performance.

5.4.2. Baselines. We compare the recommendation performances with the following approaches:

— **Popularity (POP)**: This approach provides the same recommendation list of venues to all users according to venues' popularity in the training set. Let p_j be the popularity of venue j, then $p_j = \sum_{i \in \mathbb{U}} c_{ij}^{\mathrm{Tr}}$, where c_{ij}^{Tr} is the number of check-ins performed by i at j in the training set.
— **User-based CF (UCF):** The basic idea of CF is to recommend users with venues which a group of similar users like to visit. Based on user-based CF, users' implicit preferences may be discovered by aggregating the behaviours of similar users. The

predicted rating on venue j by user i is $\hat{Q_{ij}} = \frac{\sum_{k \in \mathbb{U}, k \neq i} s(u_i, u_k) \times Q_{kj}}{\sum_{k \in \mathbb{U}, k \neq i} s(u_i, u_k)}$, where $s(u_i, u_k) = \frac{\sum_{j \in \mathbb{V}_{ik}} Q_{ij} Q_{kj}}{\sqrt{\sum_{j \in \mathbb{V}_{ik}} Q_{ij}^2} \sqrt{\sum_{j \in \mathbb{V}_{ik}} Q_{kj}^2}}$ and $\mathbb{V}_{ik}$ contains the venues visited by user i or user k.

— **SVD:** We first perform the SVD decomposition of $\mathbf{Q}$ as $\mathbf{Q} = \mathbf{USV}^T$, where $\mathbf{U} \in \mathbb{R}^{N \times k}$, $\mathbf{S} \in \mathbb{R}^{k \times k}$ and $\mathbf{V} \in \mathbb{R}^{M \times k}$. The predicted rating of user i towards venue j is then computed as $\hat{Q_{ij}} = (\mathbf{u}_i \sqrt{\mathbf{s}_i}^T) \cdot (\sqrt{\mathbf{s}_i} \mathbf{v}_j^T)$. We choose $k = 12$ by preliminary experiments.
— **Non-negative Matrix Factorization (NMF):** We use algorithm described in [Seung and Lee 2001] and empirically set the latent dimension $k = 15$.
— **Bayesian Probabilistic Matrix Factorization (BPMF):** The latent representations are learned using MCMC [Salakhutdinov and Mnih 2008a] and the prediction is computed as the inner-product of the two latent factors.
— **Bayesian Probabilistic Tensor Factorization (BPTF):** The latent representations are learned using MCMC [Xiong et al. 2010] and the prediction is computed according to Eq (6).

5.4.3. Results of Local Venue Predictions. We report here the performance comparisons in the setting of local venue prediction. Fig. 8 shows Precision@k and Recall@k of local venue predictions at each city. We observe that methods considering time factors outperform the methods that do not take time into consideration. This verifies our postulation that users do visit different kinds of venues at different times and is in line with the intuition. BPTFSLR achieves the best prediction performance in terms of precision and recall under all values of k computed, which shows the strength of considering heterogeneous inter/intra relations among users and venues in the unified model. Similar to previous studies (e.g. [Ye et al. 2011]), we observe that user-based model is a strong baseline, which beats SVD, NMF and popularity-based approaches and it is comparable to BPMF in different values of k. The use of time factors improves the performance by an average of 2.18% in recall and 8.8% in precision over the user-based model for Singapore users. The introduction of social and location regularization further improves the performance by an average of 4.67% in recall and 8.9% in precision over BPTF when considering all values of k in venue prediction for Singapore users. The corresponding improvements are 4.97%, 7%, 1.82%, 6.2% for New York City users, 1.02%, 6.37%, 9.57%, 6.49% for Chicago users and 4.55%, 2.32%, 10.58%, 7.3% for London users. In addition, we observe that performance for Singapore user is generally better than that of New York City. The reason could be that most Singapore residents perform most of check-ins in Singapore whereas New York City residents often visit areas other than New York City, which lowers the density of the training tensor.

Fig. 9 and 10 respectively show the MAP of all approaches for the four cities in two settings: time independent/dependent. We observe that methods that do not consider time factors perform badly in generating time-dependent predictions since they operate on sparser datasets where only check-ins performed during particular time periods are considered. On the other hand, approaches that consider time factors achieve much better performances. However, the performance is not as good as that of the time-independent prediction task, which is an easier problem after all. We further investigate the prediction stability in time-independent settings and observe that Bayesian approaches are generally more stable than the rest and the average standard deviation over all cities by BPTFSLR is 0.035. In time-dependent settings, we observe that predictions on weekday afternoons are the most stable while those on weekday nights are the least stable. The reason could be that people perform more regular and frequent check-ins during weekday afternoons. This observation is consistent across four cities.

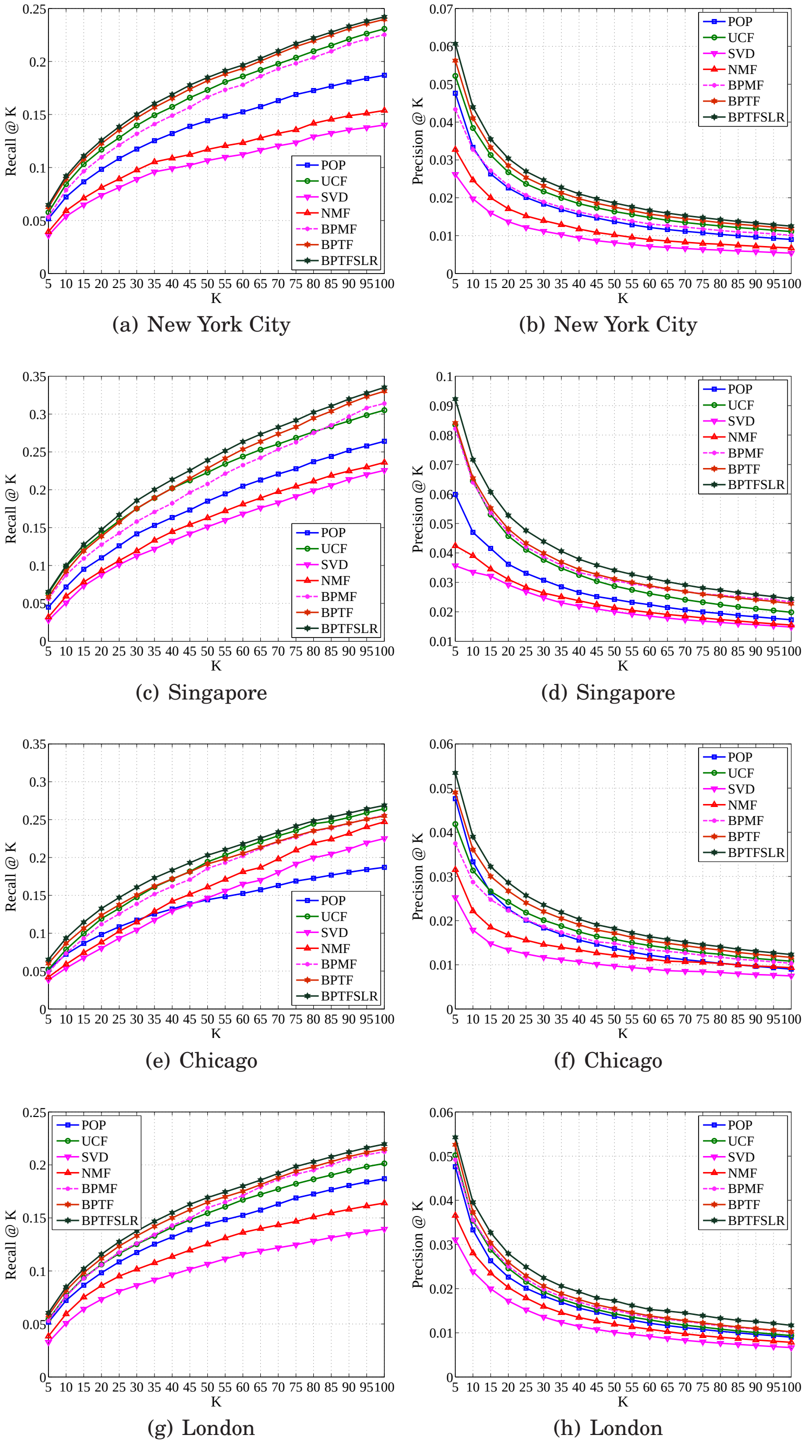


Fig. 8. Precision and Recall of Local Venue Prediction (time-independent)

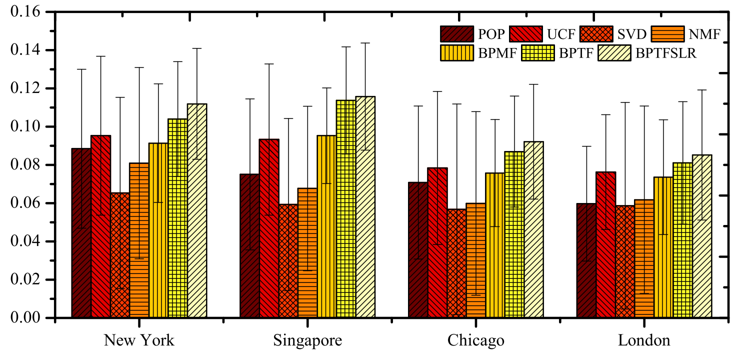


Fig. 9. MAP of Local Venue Prediction (time-independent)

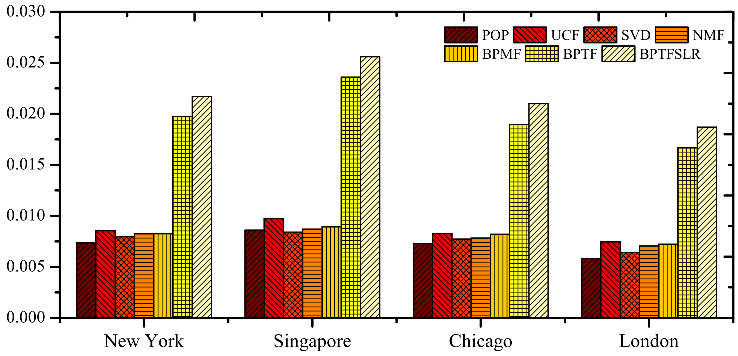


Fig. 10. MAP of Local Venue Prediction (time-dependent)

5.5. Cross Region Venue Recommendations

In this section, we report the empirical results of locally interesting venue recommendations to tourists. In this paper, we define locally interesting venues to be those that are frequently visited by local people but unknown to most of tourists. Examples include a famous food market in the neighbourhood, a nature park for local sports lovers, etc. An intuitive approach to identify a list of locally interesting venues is to cluster venues based on the proportions of local and foreign visits. First, we filter venues by categories which may not interest tourists[9]. Second, we represent each venue with proportions of unique local and foreign visitors. We next perform k-means clustering to generate three venue clusters at each geographical region. The clusters containing venues with high local visits and low foreign visits are expected to contain locally interesting venues we would like to recommend to tourists. Table IV gives some examples of locally interesting venues in four cities. We can see that most of these locally interesting venues are not available in the famous touring sites yet they are indeed of interests to tourists who like to explore local cultures. Table V further shows the statistics of mutual foreign visits of the four cities. We observe that the average percentage of locally interesting venues explored versus the total number of venues visited by tourists from

[9]Examples include bus stations, factories, post offices and etc.

Table IV. Example popular/locally interesting venues

city	popular venues
CHI	Millennium Park, Palmer House Hilton, Hyatt Regency Chicago
LDN	Harrods, London Eye, British Museum, Tower Bridge, Tower of London
NYC	Times Square, Museum of Modern Art, Madison Square Garden
SG	Clarke Quay, The Paragon, Sentosa Island, Marina Bay Sands Casino
	locally interesting venues
CHI	Food Life, Big Star, Bull & Bear, Whole Foods Market, Target, Piece Brewery & Pizzeria
LDN	Round House, Camden Stables Market, Barbican Centre, Columbia Road Flower Market
NYC	City Field, Union Square Greenmarket, Think Coffee,Bloomingdale's, Columbus Circle
SG	Old Airport Road Market & Food Centre, Bugis Junction, Newton Circus Food Centre

Table V. Statistics of cross region visiting. Each row shows the dataset of users in the corresponding city. In each table entry, numbers of venues visited/check-ins are showed on top/bottom. The corresponding numbers of locally interesting venues visited/check-ins are showed in brackets.

	Chicago	London	New York City	Singapore
CHI	—	$966(155)$ $1,590(191)$	$9,371(3,945)$ $17,752(5,862)$	$140(47)$ $206(111)$
LDN	$376(134)$ $557(165)$	—	$3,070(1,354)$ $4,251(1,621)$	$748(72)$ $1,234(111)$
NYC	$5,936(2,970)$ $16,659(5,291)$	$3,099(625)$ $9,175(895)$	—	$1,447(165)$ $1,978(341)$
SG	$135(54)$ $155(85)$	$445(30)$ $630(35)$	$1,210(320)$ $1,665(400)$	—

Chicago, London, New York City and Singapore are 30.57%, 29.72%, 27.21% and 24.4%, respectively.

5.5.1. Evaluation Strategies. The purpose of our proposed framework is to recommend tourists with personalized locally interesting venues that both match their local preferences and they are not aware of. The ideal evaluation strategy is to let the actual tourists comment on the recommended venues presented to them. However, since it is expensive and time-consuming to conduct the ideal evaluation process, we resort to an approximate evaluation by comparing the generated venue recommendation list with the actual visits by users in foreign cities. First, we investigate the prediction performance if we only recommend popular venues to tourists by generating the same recommendation list to each tourist ranked by the venues' total check-in count. Second, we evaluate the prediction performance in a personalized setting, that is, we predict a user's foreign visits based on his/her local profiles using our proposed cross region community matching. We report performance in terms of Precision@5. We compare BPTFSLR with all factor models including SVD, NMF, BPMF and BPTF. All comparisons are done in the time-independent settings. We exclude users who did not visit any foreign cities.

5.5.2. Experimental Results. Based on latent social dimensions extracted by BPTFSLR, we generate 425 local communities in Chicago, 326 in London, 495 in New York City and 434 in Singapore using AAP. Table VI and Table VII shows the results in terms of Precision@5. The first rows of the two tables list the cities in which users' foreign visits are to be predicted. The second rows list the destinations where the users' visits are to be predicted. We observe that if we recommend popular venues to tourists without considering their local profiles, the performance is comparable to that of BPMF but worse than those of BPTF and BPTFSLR. Indeed, tourists tend to visit popular venues if they are not aware of those locally interesting venues that match their personal interests. We further observe that BPTFSLR consistently outperform other state-of-the-art approaches, which again shows the positive influences by involving social factors and

Table VI. Precision@5 of venue recommendations to tourists from Chicago and London via cross region community matching.

	Chicago			London		
	LDN	NYC	SG	CHI	NYC	SG
Popularity	0.02501	0.02827	0.01935	0.01756	0.02207	0.01959
SVD	0.01659	0.02205	0.01491	0.01365	0.02002	0.01659
NMF	0.01715	0.0238	0.01596	0.01519	0.02037	0.01995
BPMF	0.02156	0.02464	0.01988	0.01687	0.0217	0.0203
BPTF	0.03045	0.03584	0.02177	0.01792	0.03206	0.03045
BPTFSLR	0.03192	0.03941	0.02632	0.02149	0.04389	0.03381

Table VII. Precision@5 of venue recommendations to tourists from New York City and Singapore via cross region community matching.

	New York City			Singapore		
	CHI	LDN	SG	CHI	LDN	NYC
Popularity	0.03567	0.02397	0.02509	0.00942	0.01915	0.02551
SVD	0.03171	0.02065	0.02009	0.00742	0.01771	0.02051
NMF	0.0343	0.02058	0.0203	0.00861	0.01876	0.02212
BPMF	0.03731	0.0224	0.0231	0.00819	0.01988	0.0245
BPTF	0.04144	0.02415	0.03171	0.0105	0.02737	0.0343
BPTFSLR	0.05012	0.0476	0.04368	0.01211	0.03402	0.03759

Table VIII. HR@5 of venue recommendations to tourists via cross region community matching using BPTFSLR (in %). The first row shows the destination cities. The first column shows the home cities of the tourists.

	Chicago	London	New York City	Singapore
CHI	—	11.25	11.38	9.8
LDN	7.5	—	12.97	11.79
NYC	15.72	14.62	—	13.1
SG	6.65	12.3	13.67	—

considering venue similarities. In addition, we observe that the predictions are more accurate when users visit cities which are geographically closer to their home cities and when the volume of check-ins performed at the target city is high. For example, the predictions for users from New York City to venues in Chicago are the most accurate as compared to other foreign visits. The reasons could be: (1) Users from these two cities may have more similar background; (2) Venues they visit may share similar local cultures; (3) Users may have better knowledge of locally interesting venues, which locate nearer to their home towns; and (4) The volume of check-ins from New York City users in Chicago is much higher than those in Singapore and London. In short, personalized recommendations considering locally interesting venues are best achieved by our proposed framework using BPTFSLR.

In order to know the proportions of users who receive at least one correct recommendation among the first few recommended venues, we compute the hit rate @ 5 (HR@5) of venue recommendations to tourists using BPTFSLR. HR@5 shows the proportion of tourists who receive at least one correct recommendation among the first five recommended venues. Table VIII shows the results. Again, the predictions are more accurate when users visit cities which are geographically closer to their home cities and when the volume of check-ins performed at the target city is high. The highest HR@5 is obtained from recommending Chicago venues to tourists from New York City. It is worth mentioning that both Precision@5 and HR@5 are not high in their absolute values in this experiment, which is due to the low density of the user-venue-time tensor, whose density is currently in the order of 10^{-5}. And the performances are expected to improve when the number of LBSNs' users continues to grow and more check-in activities are recorded.

5.6. Summary

We summarize the experimental results as follows.

(1) The introduction of time factors is shown to be able to improve the venue prediction accuracy in both local and foreign venue predictions.
(2) The social and venue regularization leads to further improvements on the recommendation performance.
(3) Our proposed BPTFSLR gives the best recommendation accuracy through cross region community matching.

6. RELATED WORK

Our work is related to three main research areas: (1) location recommendation, (2) travel recommendation and (3) latent factor models.

6.1. Location Recommendation

The recent boom of LBSNs has motivated emerging research on point-of-interest (POI) or more generally location recommendations [Ye et al. 2011; Baltrunas et al. 2011; Ying et al. 2012; Zhou et al. 2012]. Location recommendation aims to recommend a list of POIs or locations to a user based on the user's past visiting histories. These lines of work usually focus on general recommendation tasks in a traditional CF framework. For example, Ye et al. compared the influences on user similarity that were based on historical behaviour, geographical distance and friend network in POI recommendation task [Ye et al. 2011]. Ying et al. proposed to consider both user preferences and location properties in their recommendation framework [Ying et al. 2012]. Recently, Zhou et al. studied and compared the performances of different CF recommenders, including user-based, item-based and probabilistic latent semantic analysis in location recommendation, where they reported that the probabilistic approach gave the optimum performance [Zhou et al. 2012]. From another angle, Baltrunas et al. introduced a context-aware recommender system for POIs, where the system considered contextual factors such as distance to POIs, temperature, users' mood, etc [Baltrunas et al. 2011]. There are two main differences between our work and these related work: (1) We study a new problem which aims to provide tourists with recommendations based on their local visits and (2) None of these work has studied the effects of simultaneously considering time, social relations and venue similarities.

6.2. Travel Recommendation

In Web 2.0 communities, people often share their traveling experience in blogs, forums and social networks in terms of travelogues, photos, etc. These geo-referenced media resources contain rich information of tourism, which motivates research on generating travel recommendations from these user generated contents [Gao et al. 2010; Hao et al. 2010; Cheng et al. 2011b; Zhao et al. 2011; Lucchese et al. 2012]. Gao et al. presented a travel guidance system, which automatically recognized and ranked the landmarks for travellers from Flickr photos and Yahoo Travel guide [Gao et al. 2010]. Hao et al. proposed a location-topic model to model travelogue documents and develop a tour destination recommendation [Hao et al. 2010]. To recommend a destination, a user needs to issue a query and then the system utilizes the topic model to select a destination with highest matching score. Cheng et al. leveraged community-contributed photos from Flickr to provide personalized travel recommendation based on people's attributes, such as gender, race and age in a probabilistic Bayesian learning framework [Cheng et al. 2011b]. More recently, Lucchese et al. proposed an interactive random walk approach of personalized recommendations of touristic places based on knowledge mined from Flickr and Wikipedia [Lucchese et al. 2012]. While these efforts all aim to provide personalized recommendations of touristic points based on users' past behaviours or destinations' popularity, our work focuses on recommending locally interesting venues and aims to solve a problem of cross region recommendation. In addition, we utilize user generated location contents in LBSNs, which better connect the physical world with the online virtual world.

6.3. Latent Factor Models

Latent factor models are shown to be promising in recommendation tasks such as Netflix competition [Bell et al. 2007], results diversification [Shi et al. 2012], review helpfulness prediction [Moghaddam et al. 2012] and web site recommendations [Ma et al. 2011]. The underlying assumption of using latent factor models is that the entities, such as users, items (venues, reviews, products, etc) can be modeled by a set of latent representations, which together determine the preferences of unknown items in a probabilistic way. For example, Moghaddam et al. proposed a series of increasingly sophisticated probabilistic graphical models based on tensor factorization and showed their effectiveness in the prediction of review helpfulness [Moghaddam et al. 2012]. Recent work by Cheng et al. has shown a positive influence by introducing social regularization in POI recommendations performed on Gowalla [Cheng et al. 2012]. Our proposed framework differs from these efforts in two main aspects: (1) The framework considers temporal changes of users' preferences and heterogeneous intra/inter entity relations in a unified manner; and (2) We derive a Bayesian treatment to sample latent factors, which both avoids overfitting and tedious parameter tuning.

7. CONCLUSIONS AND FUTURE WORK

This paper identified and studied a new problem of personalized locally interesting venue recommendation to tourists. We proposed an effective framework for community-centric latent social dimensions extraction by taking into consideration heterogenous relations among users, venues and time. With the detected local communities, we then utilized cross region community matching to generate locally interesting venues to tourists. Experimental results have well verified the quality of the extracted latent social dimensions and the effectiveness of our proposed framework in cross region recommendations. For future work, we will look into how to make better use of local/foreign friends relations across different regions. In addition, we will develop an incremental approach to make community profiles adaptable to the evolving characteristics of users' interests in both long term and short term to further improve the recommendation accuracy.

REFERENCES

Kamal Ali and Wijnand van Stam. 2004. TiVo: making show recommendations using a distributed collaborative filtering architecture. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04)*. ACM, New York, NY, USA, 394–401.

Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. 2003. An introduction to MCMC for machine learning. *Machine learning* 50, 1-2 (2003), 5–43.

Linas Baltrunas, Bernd Ludwig, Stefan Peer, and Francesco Ricci. 2011. Context-Aware Places of Interest Recommendations for Mobile Users. In *Design, User Experience, and Usability. Theory, Methods, Tools and Practice*, Aaron Marcus (Ed.). Lecture Notes in Computer Science, Vol. 6769. Springer Berlin Heidelberg, 531–540.

Robert Bell, Yehuda Koren, and Chris Volinsky. 2007. Modeling relationships at multiple scales to improve accuracy of large recommender systems. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '07)*. ACM, New York, NY, USA, 95–104.

Robert M. Bell and Yehuda Koren. 2007. Lessons from the Netflix prize challenge. *SIGKDD Explorations Newsletter* 9, 2 (Dec. 2007), 75–79.

Léon Bottou. 2004. Stochastic learning. In *Advanced lectures on machine learning*. Springer, 146–168.

An-Jung Cheng, Yan-Ying Chen, Yen-Ta Huang, Winston H. Hsu, and Hong-Yuan Mark Liao. 2011b. Personalized travel recommendation by mining people attributes from community-contributed photos. In *Proceedings of the 19th ACM international conference on Multimedia (MM '11)*. ACM, New York, NY, USA, 83–92.

Chen Cheng, Haiqin Yang, Irwin King, and Michael R Lyu. 2012. Fused matrix factorization with geographical and social influence in location-based social networks. In *AAAI'12: Proceedings of Twenty-Sixth Conference on Artificial Intelligence*.

Zhiyuan Cheng, James Caverlee, Kyumin Lee, and Daniel Z Sui. 2011a. Exploring millions of footprints in location sharing services. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Eunjoon Cho, Seth A. Myers, and Jure Leskovec. 2011. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '11)*. ACM, New York, NY, USA, 1082–1090.

Nathan Eagle and Alex Sandy Pentland. 2009. Eigenbehaviors: Identifying structure in routine. *Behavioral Ecology and Sociobiology* 63, 7 (2009), 1057–1066.

Brendan J Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *science* 315, 5814 (2007), 972–976.

Yue Gao, Jinhui Tang, Richang Hong, Qionghai Dai, Tat-Seng Chua, and Ramesh Jain. 2010. W2Go: a travel guidance system by automatic landmark ranking. In *Proceedings of the international conference on Multimedia (MM '10)*. ACM, New York, NY, USA, 123–132.

Alan E Gelfand and Adrian FM Smith. 1990. Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association* 85, 410 (1990), 398–409.

F. Girardin, F. Calabrese, F.D. Fiore, C. Ratti, and J. Blat. 2008. Digital Footprinting: Uncovering Tourists with User-Generated Content. *Pervasive Computing, IEEE* 7, 4 (2008), 36–43.

David Goldberg, David Nichols, Brian M. Oki, and Douglas Terry. 1992. Using collaborative filtering to weave an information tapestry. *Commun. ACM* 35, 12 (Dec. 1992), 61–70.

Qiang Hao, Rui Cai, Changhu Wang, Rong Xiao, Jiang-Ming Yang, Yanwei Pang, and Lei Zhang. 2010. Equip tourists with knowledge mined from travelogues. In *Proceedings of the 19th international conference on World wide web (WWW '10)*. ACM, New York, NY, USA, 401–410.

Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. 1999. An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '99)*. ACM, New York, NY, USA, 230–237.

Patrik O. Hoyer. 2004. Non-negative Matrix Factorization with Sparseness Constraints. *J. Mach. Learn. Res.* 5 (Dec. 2004), 1457–1469.

G. Linden, B. Smith, and J. York. 2003. Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE* 7, 1 (2003), 76–80.

Claudio Lucchese, Raffaele Perego, Fabrizio Silvestri, Hossein Vahabi, and Rossano Venturini. 2012. How random walks can help tourism. In *Proceedings of the 34th European conference on Advances in Information Retrieval (ECIR'12)*. Springer-Verlag, Berlin, Heidelberg, 195–206.

Hao Ma, Chao Liu, Irwin King, and Michael R. Lyu. 2011. Probabilistic factor models for web site recommendation. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 265–274.

Miller McPherson, Lynn Smith-Lovin, and James M Cook. 2001. Birds of a feather: Homophily in social networks. *Annual review of sociology* (2001), 415–444.

Samaneh Moghaddam, Mohsen Jamali, and Martin Ester. 2012. ETF: extended tensor factorization model for personalizing prediction of review helpfulness. In *Proceedings of the fifth ACM international conference on Web search and data mining (WSDM '12)*. ACM, New York, NY, USA, 163–172.

A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. 2011. An empirical study of geographic user activity patterns in foursquare. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Ruslan Salakhutdinov and Andriy Mnih. 2008a. Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. In *Proceedings of the 25th international conference on Machine learning (ICML '08)*. ACM, New York, NY, USA, 880–887.

Ruslan Salakhutdinov and Andriy Mnih. 2008b. Probabilistic matrix factorization. *Advances in neural information processing systems* 20 (2008), 1257–1264.

D Seung and L Lee. 2001. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems* 13 (2001), 556–562.

Yue Shi, Xiaoxue Zhao, Jun Wang, Martha Larson, and Alan Hanjalic. 2012. Adaptive diversification of recommendation results via latent factor portfolio. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval (SIGIR '12)*. ACM, New York, NY, USA, 175–184.

Lei Tang and Huan Liu. 2009. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '09)*. ACM, New York, NY, USA, 817–826.

Kaijun Wang, Junying Zhang, Dan Li, Xinna Zhang, and Tao Guo. 2008. Adaptive affinity propagation clustering. *arXiv preprint arXiv:0805.1096* (2008).

J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, and Yi Ma. 2009. Robust Face Recognition via Sparse Representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31, 2 (2009), 210–227.

Liang Xiong, Xi Chen, Tzu-Kuo Huang, Jeff Schneider, and Jaime G Carbonell. 2010. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of SIAM Data Mining*, Vol. 2010.

Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee. 2011. Exploiting geographical influence for collaborative point-of-interest recommendation. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 325–334.

Josh Jia-Ching Ying, Eric Hsueh-Chan Lu, Wen-Ning Kuo, and Vincent S. Tseng. 2012. Urban point-of-interest recommendation by mining user check-in behaviors. In *Proceedings of the ACM SIGKDD International Workshop on Urban Computing (UrbComp '12)*. ACM, New York, NY, USA, 63–70.

Ted Tao Yuan, Zhaohui Chen, and Mike Mathieson. 2011. Predicting eBay listing conversion. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 1335–1336.

Yi-Liang Zhao, Qiang Chen, Shuicheng Yan, Daqing Zhang, and Tat-Seng Chua. 2013. Detecting Profilable and Overlapping Communities with User-generated Multimedia Contents in LBSNs. *ACM Trans. Multimedia Comput. Commun. Appl.* (2013), (To appear).

Yi-Liang Zhao, Yan-Tao Zheng, Xiangdong Zhou, and Tat-Seng Chua. 2011. Generating Representative Views of Landmarks via Scenic Theme Detection. In *Advances in Multimedia Modeling*. Lecture Notes in Computer Science, Vol. 6523. Springer Berlin Heidelberg, 392–402.

Dequan Zhou, Bin Wang, SeyyedMohammadreza Rahimi, and Xin Wang. 2012. A Study of Recommending Locations on Location-Based Social Network by Collaborative Filtering. In *Advances in Artificial Intelligence*, Leila Kosseim and Diana Inkpen (Eds.). Lecture Notes in Computer Science, Vol. 7310. Springer Berlin Heidelberg, 255–266.

Received April 2013; revised July 2013; accepted September 2013

Online Appendix to:
Personalized Recommendations of Locally Interesting Venues to Tourists via Cross Region Community Matching

YI-LIANG ZHAO, National University of Singapore
LIQIANG NIE, National University of Singapore
XIANGYU WANG, National University of Singapore
TAT-SENG CHUA, National University of Singapore

A. CONDITIONAL DISTRIBUTION IN GIBBS SAMPLING

In this section, we give the updated conditional distributions used in Algorithm 1 in the paper. According to the graphical model shown in Fig. 3 in the paper, the joint posterior distribution can be factorized as:

$$\begin{aligned}
&p(\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}, \tau_Q, \tau_B, \tau_R, \Theta_U, \Theta_V, \Theta_S, \Theta_D, \Theta_T|\mathbf{Q}, \mathbf{R}, \mathbf{B}) \\
&\propto p(\mathbf{Q}|\mathbf{U}, \mathbf{V}, \mathbf{T}, \tau_Q)p(\mathbf{R}|\mathbf{U}, \mathbf{S}, \tau_R)p(\mathbf{B}|\mathbf{V}, \mathbf{D}, \tau_B) \\
&p(\mathbf{U}|\Theta_U)p(\mathbf{V}|\Theta_V)p(\mathbf{S}|\Theta_S)p(\mathbf{D}|\Theta_D)p(\mathbf{T}|\Theta_T) \\
&p(\tau_Q)p(\tau_B)p(\tau_R)p(\Theta_U)p(\Theta_V)p(\Theta_S)p(\Theta_D)p(\Theta_T)
\end{aligned} \tag{21}$$

By plugging in all the model components described in Section 3.3.4 in the paper and carrying out marginalization for each variable, we derive the conditional distributions in the following subsections for hyperparameters and model parameters, respectively.

A.1. Conditional Distributions of Hyperparameters

In this section, we give details of conditional distributions of hyperparameters, which include precision variables and model parameters for latent variables.

Precision Variables

(1) τ_Q

By using the conjugate prior for the precision τ_Q, we have that the conditional distribution of τ_Q given $\mathbf{Q}, \mathbf{U}, \mathbf{V}, \mathbf{T}$ also follows the Wishart distribution:

$$p(\tau_Q|\mathbf{Q}, \mathbf{U}, \mathbf{V}, \mathbf{T}) = \mathcal{W}(\tau_Q|W_1^*, v_1^*), \tag{22}$$

where W_1^* and v_1^* are the parameters in the posterior distribution and updated as follows.

$$\begin{cases}(W_1^*)^{-1} = W_1^{-1} + \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^T I_{ij}^k (Q_{ij}^k - \langle \mathbf{u}_i, \mathbf{v}_j, \mathbf{t}_k \rangle)^2, \\ v_1^* = v_1 + \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^T I_{ij}^k. \end{cases}$$

(2) τ_R

Similarly, we derive the conditional distribution of τ_R given $\mathbf{S}, \mathbf{U}, \mathbf{R}$ as follows.

$$p(\tau_R|\mathbf{R}, \mathbf{U}, \mathbf{S}) = \mathcal{W}(\tau_R|W_1^*, v_1^*), \tag{23}$$

DOI:http://dx.doi.org/10.1145/0000000.0000000

where

$$\begin{cases}(W_1^*)^{-1} = W_1^{-1} + \sum_{r=1}^N \sum_{i=1}^N I_{ri}^R (R_{ri} - \mathbf{s}_r^T \mathbf{u}_i)^2, \\ v_1^* = v_1 + \sum_{r=1}^N \sum_{i=1}^N I_{ri}^R. \end{cases} \tag{24}$$

(3) τ_B
And the conditional distribution of τ_B given $\mathbf{B}, \mathbf{V}, \mathbf{D}$ is:

$$p(\tau_B|\mathbf{B}, \mathbf{V}, \mathbf{D}) = \mathcal{W}(\tau_B|W_1^*, v_1^*), \tag{25}$$

where

$$\begin{cases}(W_1^*)^{-1} = W_1^{-1} + \sum_{j=1}^M \sum_{l=1}^M I_{jl}^B (B_{jl} - \mathbf{v}_j^T \mathbf{d}_l)^2, \\ v_1^* = v_1 + \sum_{j=1}^M \sum_{l=1}^M I_{jl}^B. \end{cases} \tag{26}$$

Hyperparameters for Model Variables
Next, we work out the conditional distribution for $\Theta_U \equiv \{\boldsymbol{\mu}_U, \boldsymbol{\Lambda}_U\}$, $\Theta_V \equiv \{\boldsymbol{\mu}_V, \boldsymbol{\Lambda}_V\}$, $\Theta_S \equiv \{\boldsymbol{\mu}_S, \boldsymbol{\Lambda}_S\}$, $\Theta_D \equiv \{\boldsymbol{\mu}_D, \boldsymbol{\Lambda}_D\}$ and $\Theta_T \equiv \{\boldsymbol{\mu}_T, \boldsymbol{\Lambda}_T\}$.

(1) Θ_U
Θ_U is conditionally independent on all the other parameters given $\mathbf{U}$. We thus integrate out all the random variables in Eq (21) except $\mathbf{U}$ and obtain the Gaussian-Wishart distribution.

$$p(\Theta_U|\mathbf{U}) = \mathcal{N}(\boldsymbol{\mu}_U|\boldsymbol{\mu}_0^*, (\beta^*\boldsymbol{\Lambda}_U)^{-1})\mathcal{W}(\boldsymbol{\Lambda}_U|\mathbf{W}_0^*, v_0^*), \tag{27}$$

where the parameters are updated as follows.

$$\begin{cases}\boldsymbol{\mu}_0^* = \frac{\beta\boldsymbol{\mu}_0 + N\bar{\mathbf{u}}}{\beta+N}, \ \ \beta^* = \beta + N, \ \ v_0^* = v_0 + N, \\ (\mathbf{W}_0^*)^{-1} = \mathbf{W}_0^{-1} + N\boldsymbol{\Phi} + \frac{\beta N}{\beta+N}(\boldsymbol{\mu}_0 - \bar{\mathbf{u}})(\boldsymbol{\mu}_0 - \bar{\mathbf{u}})^T. \end{cases} \tag{28}$$

where $\bar{\mathbf{u}} = \frac{1}{N}\sum_{i=1}^N \mathbf{u}_i$ and $\boldsymbol{\Phi} = \frac{1}{N}\sum_{i=1}^N (\mathbf{u}_i - \bar{\mathbf{u}})(\mathbf{u}_i - \bar{\mathbf{u}})^T$.

(2) Θ_V
Similarly, we can get the conditional distribution for Θ_V as follows.

$$p(\Theta_V|\mathbf{V}) = \mathcal{N}(\boldsymbol{\mu}_V|\boldsymbol{\mu}_0^*, (\beta^*\boldsymbol{\Lambda}_V)^{-1})\mathcal{W}(\boldsymbol{\Lambda}_V|\mathbf{W}_0^*, v_0^*), \tag{29}$$

where

$$\begin{cases}\boldsymbol{\mu}_0^* = \frac{\beta\boldsymbol{\mu}_0 + M\bar{\mathbf{v}}}{\beta+M}, \ \ \beta^* = \beta + M, \ \ v_0^* = v_0 + M, \\ (\mathbf{W}_0^*)^{-1} = \mathbf{W}_0^{-1} + M\boldsymbol{\Phi} + \frac{\beta M}{\beta+M}(\boldsymbol{\mu}_0 - \bar{\mathbf{v}})(\boldsymbol{\mu}_0 - \bar{\mathbf{v}})^T. \end{cases} \tag{30}$$

where $\bar{\mathbf{v}} = \frac{1}{M}\sum_{j=1}^M \mathbf{v}_j$ and $\boldsymbol{\Phi} = \frac{1}{N}\sum_{j=1}^M (\mathbf{v}_i - \bar{\mathbf{v}})(\mathbf{v}_j - \bar{\mathbf{v}})^T$.

(3) Θ_S
And the conditional distribution for Θ_S is:

$$p(\Theta_S|\mathbf{S}) = \mathcal{N}(\boldsymbol{\mu}_S|\boldsymbol{\mu}_0^*, (\beta^*\boldsymbol{\Lambda}_S)^{-1})\mathcal{W}(\boldsymbol{\Lambda}_S|\mathbf{W}_0^*, v_0^*), \tag{31}$$

where

$$\begin{cases} \boldsymbol{\mu}_0^* = \frac{\beta\boldsymbol{\mu}_0 + N\bar{\mathbf{s}}}{\beta+N}, \quad \beta^* = \beta + N, \quad v_0^* = v_0 + N, \\ (\mathbf{W}_0^*)^{-1} = \mathbf{W}_0^{-1} + N\boldsymbol{\Phi} + \frac{\beta N}{\beta+N}(\boldsymbol{\mu}_0 - \bar{\mathbf{s}})(\boldsymbol{\mu}_0 - \bar{\mathbf{s}})^T. \end{cases} \tag{32}$$

where $\bar{\mathbf{s}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{s}_i$ and $\boldsymbol{\Phi} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{s}_i - \bar{\mathbf{s}})(\mathbf{s}_i - \bar{\mathbf{s}})^T$.

(4) Θ_D
The conditional distribution for Θ_D is:

$$p(\Theta_D|\mathbf{D}) = \mathcal{N}(\boldsymbol{\mu}_D|\boldsymbol{\mu}_0^*, (\beta^*\boldsymbol{\Lambda}_D)^{-1})\mathcal{W}(\boldsymbol{\Lambda}_D|\mathbf{W}_0^*, v_0^*), \tag{33}$$

where

$$\begin{cases} \boldsymbol{\mu}_0^* = \frac{\beta\boldsymbol{\mu}_0 + M\bar{\mathbf{d}}}{\beta+M}, \quad \beta^* = \beta + M, \quad v_0^* = v_0 + M, \\ (\mathbf{W}_0^*)^{-1} = \mathbf{W}_0^{-1} + M\boldsymbol{\Phi} + \frac{\beta M}{\beta+M}(\boldsymbol{\mu}_0 - \bar{\mathbf{d}})(\boldsymbol{\mu}_0 - \bar{\mathbf{d}})^T. \end{cases}$$

where $\bar{\mathbf{d}} = \frac{1}{M}\sum_{j=1}^{M}\mathbf{v}_j$ and $\boldsymbol{\Phi} = \frac{1}{N}\sum_{j=1}^{M}(\mathbf{d}_i - \bar{\mathbf{d}})(\mathbf{d}_i - \bar{\mathbf{d}})^T$.

(5) Θ_T
Finally, the conditional distribution of Θ_T also follows the Gaussian-Wishart distribution:

$$p(\Theta_T|\mathbf{T}) = \mathcal{N}(\boldsymbol{\mu}_T|\boldsymbol{\mu}_1^*, (\beta^*\boldsymbol{\Lambda}_T)^{-1})\mathcal{W}(\boldsymbol{\Lambda}_T|\mathbf{W}_0^*, v_0^*), \tag{34}$$

where

$$\begin{cases} \boldsymbol{\mu}_1^* = \frac{\beta\boldsymbol{\mu}_1 + \mathbf{t}_1}{\beta+1}, \quad \beta^* = \beta + 1, \quad v_0^* = v_0 + T, \\ (\mathbf{W}_0^*)^{-1} = \mathbf{W}_0^{-1} + \sum_{k=2}^{T}(\mathbf{t}_k - \mathbf{t}_{k-1})(\mathbf{t}_k - \mathbf{t}_{k-1})^T + \frac{\beta}{\beta+1}(\mathbf{t}_1 - \boldsymbol{\mu}_1)(\mathbf{t}_1 - \boldsymbol{\mu}_1)^T. \end{cases} \tag{35}$$

A.2. Conditional Distributions of Model Variables

In this section, we give details of conditional distributions of model parameters: $\mathbf{U}, \mathbf{V}, \mathbf{S}, \mathbf{D}, \mathbf{T}$.
(1) $\mathbf{U}$
The conditional distribution of $\mathbf{U}$ can be factorized with respect to each individual user:

$$p(\mathbf{U}|\mathbf{Q}, \mathbf{V}, \mathbf{T}, \mathbf{R}, \mathbf{S}, \tau_Q, \tau_R, \Theta_U) = \prod_{i=1}^{N} p(\mathbf{u}_i|\mathbf{Q}, \mathbf{V}, \mathbf{T}, \mathbf{R}, \mathbf{S}, \tau_Q, \tau_R, \Theta_U) \tag{36}$$

and

$$p(\mathbf{u}_i|\mathbf{Q}, \mathbf{V}, \mathbf{T}, \mathbf{R}, \mathbf{S}, \tau_Q, \tau_R, \Theta_U) = \mathcal{N}(\mathbf{u}_i|\boldsymbol{\mu}_i^*, (\boldsymbol{\Lambda}_i^*)^{-1}), \tag{37}$$

where

$$\begin{cases} \boldsymbol{\mu}_i^* = (\boldsymbol{\Lambda}_i^*)^{-1}(\boldsymbol{\Lambda}_U\boldsymbol{\mu}_U + \tau_Q\sum_{j=1}^{M}\sum_{k=1}^{T} I_{ij}^k Q_{ij}^k \mathbf{a}_{jk} + \tau_R\sum_{r=1}^{N} I_{ri}^R R_{ri}\mathbf{s}_r), \\ \boldsymbol{\Lambda}_i^* = \boldsymbol{\Lambda}_U + \tau_Q\sum_{j=1}^{M}\sum_{k=1}^{T} I_{ij}^k \mathbf{a}_{jk}\mathbf{a}_{jk}^T + \tau_R\sum_{r=1}^{N} I_{ri}^R \mathbf{s}_r\mathbf{s}_r^T. \end{cases}$$

where $\mathbf{a}_{jk} = \mathbf{v}_j \circ \mathbf{t}_k$ is the element-wise product of $\mathbf{v}_j$ and $\mathbf{t}_k$.

(2) **V**

Similarly, the conditional distribution of **V** can be factorized with respect to each venue as follows.

$$p(\mathbf{v}_j|\mathbf{Q}, \mathbf{U}, \mathbf{T}, \mathbf{B}, \mathbf{D}, \tau_Q, \tau_B, \Theta_V) = \mathcal{N}(\mathbf{v}_j|\boldsymbol{\mu}_j^*, (\boldsymbol{\Lambda}_j^*)^{-1}), \tag{38}$$

where

$$\begin{cases} \boldsymbol{\mu}_j^* = (\boldsymbol{\Lambda}_j^*)^{-1}(\boldsymbol{\Lambda}_V \boldsymbol{\mu}_V + \tau_Q \sum_{i=1}^N \sum_{k=1}^T I_{ij}^k Q_{ij}^k \mathbf{f}_{ik} + \tau_B \sum_{l=1}^M I_{jl}^B B_{jl} \mathbf{d}_l), \\ \boldsymbol{\Lambda}_j^* = \boldsymbol{\Lambda}_V + \tau_Q \sum_{i=1}^N \sum_{k=1}^T I_{ij}^k \mathbf{f}_{ik} \mathbf{f}_{ik}^T + \tau_B \sum_{l=1}^M I_{jl}^B \mathbf{d}_l \mathbf{d}_l^T. \end{cases}$$

where $\mathbf{f}_{ik} = \mathbf{u}_i \circ \mathbf{t}_k$ is the element-wise product of $\mathbf{u}_i$ and $\mathbf{t}_k$.

(3) **S**

And the conditional distribution of $\mathbf{s}_r$ is:

$$p(\mathbf{s}_r|\mathbf{R}, \mathbf{U}, \tau_R, \Theta_S) = \mathcal{N}(\mathbf{s}_r|\boldsymbol{\mu}_r^*, (\boldsymbol{\Lambda}_r^*)^{-1}), \tag{39}$$

where

$$\begin{cases} \boldsymbol{\mu}_r^* = (\boldsymbol{\Lambda}_r^*)^{-1}(\boldsymbol{\Lambda}_S \boldsymbol{\mu}_S + \tau_R \sum_{i=1}^N I_{ri}^R R_{ri} \mathbf{u}_i), \\ \boldsymbol{\Lambda}_r^* = \boldsymbol{\Lambda}_S + \tau_R \sum_{i=1}^N I_{ri}^R \mathbf{u}_i \mathbf{u}_i^T \end{cases}$$

(4) **D**

The conditional distribution of $\mathbf{d}_l$ is

$$p(\mathbf{d}_l|\mathbf{B}, \mathbf{V}, \tau_B, \Theta_D) = \mathcal{N}(\mathbf{d}_l|\boldsymbol{\mu}_l^*, (\boldsymbol{\Lambda}_l^*)^{-1}), \tag{40}$$

where

$$\begin{cases} \boldsymbol{\mu}_l^* = (\boldsymbol{\Lambda}_l^*)^{-1}(\boldsymbol{\Lambda}_D \boldsymbol{\mu}_D + \tau_B \sum_{j=1}^M I_{jl}^B B_{jl} \mathbf{v}_j), \\ \boldsymbol{\Lambda}_l^* = \boldsymbol{\Lambda}_D + \tau_B \sum_{j=1}^M I_{jl}^B \mathbf{v}_j \mathbf{v}_j^T \end{cases}$$

(5) **T**

Finally, the conditional distribution of $\mathbf{t}_k$ also follows the Gaussian distribution as follows.

$$p(\mathbf{t}_k|\mathbf{Q}, \mathbf{U}, \mathbf{V}, \mathbf{t}_{-k}, \tau_Q, \Theta_T) = \mathcal{N}(\mathbf{t}_k|\boldsymbol{\mu}_k^*, (\boldsymbol{\Lambda}_k^*)^{-1}), \tag{41}$$

where $\mathbf{t}_{-k}$ denotes all the time feature vectors except $\mathbf{t}_k$. The mean vectors and the precision matrices are updated as follows.

$$\boldsymbol{\mu}_k^* = \begin{cases} \frac{\mathbf{t}_2+\boldsymbol{\mu}_T}{2} & \text{if } k = 1 \\ (\boldsymbol{\Lambda}_k^*)^{-1}\left[\boldsymbol{\Lambda}_T(\mathbf{t}_{k-1}+\mathbf{t}_{k+1}) + \tau_Q \sum_{i=1}^N \sum_{j=1}^M I_{ij}^k Q_{ij}^k \mathbf{x}_{ij}\right] & \text{if } 2 \le k < T \\ (\boldsymbol{\Lambda}_k^*)^{-1}(\boldsymbol{\Lambda}_T \mathbf{t}_{k-1} + \tau_Q \sum_{i=1}^N \sum_{j=1}^M I_{ij}^k Q_{ij}^k \mathbf{x}_{ij}) & \text{if } k = T \end{cases}$$

$$\boldsymbol{\Lambda}_k^* = \begin{cases} 2\boldsymbol{\Lambda}_T + \tau_Q \sum_{i=1}^N \sum_{j=1}^M I_{ij}^k \mathbf{x}_{ij}\mathbf{x}_{ij}^T & \text{if } k < T \\ \boldsymbol{\Lambda}_T + \tau_Q \sum_{i=1}^N \sum_{j=1}^M I_{ij}^k \mathbf{x}_{ij}\mathbf{x}_{ij}^T & \text{if } k = T \end{cases}$$

where $\mathbf{x}_{ij} = \mathbf{u}_i \circ \mathbf{v}_j$ is the element-wise product of $\mathbf{u}_i$ and $\mathbf{v}_j$.